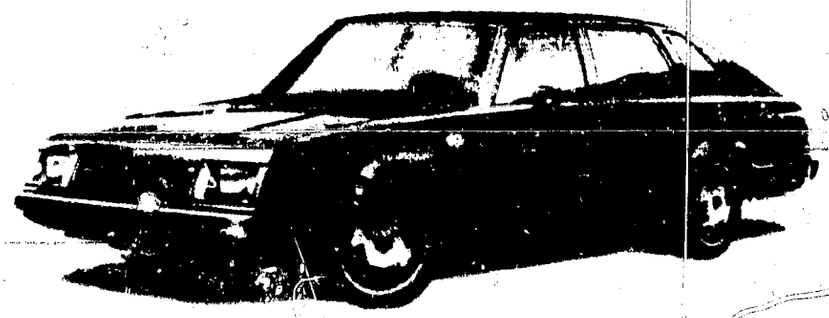
9750-334

NEAR-TERM HYBRID VEHICLE PROGRAM

FINAL REPORT -- PHASE I

Appendix B — Design Trade-Off Studies Report

Volume III — Computer Program Listings



Contract No. 955190

Submitted to

Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, California 91103

Submitted by

General Electric Company
Corporate Research and Development
Schenectady, New York 12301

October 8, 1979

GENERAL ELECTRIC

SRD-79-134/5

(NASA-CR-163229) NEAR-TERM HYBRID VEHICLE PROGRAM, PHASE 1. APPENDIX B: DESIGN TRADE-OFF STUDIES REPORT. VOLUME 3: COMPUTER PROGRAM LISTINGS Final Report (General Electric Co.) 115 p HC A06/MF A01 CSCL 13F G3/85 22357

N80-26205

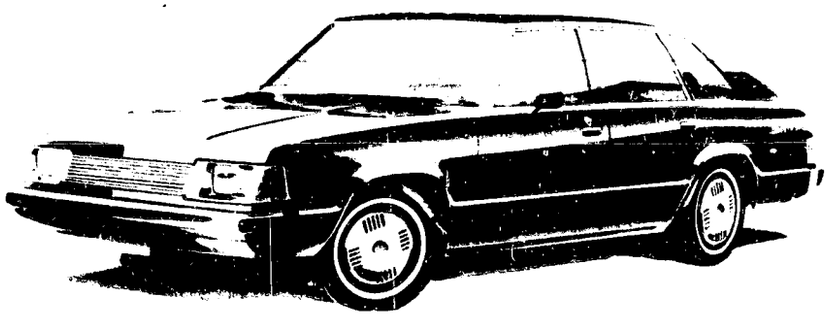
Unclas

NEAR-TERM HYBRID VEHICLE PROGRAM

FINAL REPORT — PHASE I

Appendix B — Design Trade-Off Studies Report

Volume III — Computer Program Listings



Contract No. 955190

Submitted to

Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, California 91103

Submitted by

General Electric Company
Corporate Research and Development
Schenectady, New York 12301

October 8, 197[illegible]

GENERAL ELECTRIC

FOREWORD

The Electric and Hybrid Vehicle (EHV) Program was established in DOE in response to the Electric and Hybrid Vehicle Research, Development, and Demonstration Act of 1976. Responsibility for the EHV Program resides in the Office of Electric and Hybrid Vehicle Systems of DOE. The Near-Term Hybrid Vehicle (NTHV) Program is an element of the EHV Program. DOE has assigned procurement and management responsibility for the Near-Term Hybrid Vehicle Program to JPL.

The overall objective of the DOE EHV Program is to promote the development of electric and hybrid vehicle technologies and to demonstrate the validity of these systems as transportation options which are less dependent on petroleum resources.

As part of the NTHV Program, General Electric and its subcontractors have completed studies leading to the Preliminary Design of a hybrid passenger vehicle which is projected to have the maximum potential for reducing petroleum consumption in the near term (commencing in 1985). This work has been done under JPL Contract 955190, Modification 3, Phase I of the Near-Term Hybrid Vehicle Program.

This volume is part of Deliverable Item 7, Final Report, of the Phase I studies. In accordance with Data Requirement Description 7 of the Contract, the following documents are submitted as appendices:

APPENDIX A is the Mission Analysis and Performance Specification Studies Report that constitutes Deliverable Item 1 and reports on the work of Task 1.

APPENDIX B is a three-volume set that constitutes Deliverable Item 2 and reports on the work of Task 2. The three volumes are:

- Volume I -- Design Trade-Off Studies Report
- Volume II -- Supplement to Design Trade-Off Studies Report, Volume I
- Volume III -- Computer Program Listings.

APPENDIX C is the Preliminary Design Data Package that constitutes Deliverable Item 3 and reports on the work of Task 3.

APPENDIX D is the Sensitivity Analysis Report that constitutes Deliverable Item 8 and reports on Task 4.

The three classifications - Appendix, Deliverable Item, and Task number - may be used interchangeably in these documents. The interrelationship is tabulated below:

Appendix	Deliverable Item	Task	Title
A	1	1	Mission Analysis and Performance Specification Studies Report
B	2	2	Vol. I - Design Trade-Off Studies Report
			Vol. II - Supplement to Design Trade-Off Studies Report
			Vol. III - Computer Program Listings
C	3	3	Preliminary Design Data Package
D	8	4	Sensitivity Analysis Report

This is Volume III, Computer Program Listings, of Appendix B. It presents the Hybrid Vehicle Design Program (HYVELD) and the Hybrid Vehicle Simulation Program (HYVEC).

TABLE OF CONTENTS

Section		Page
1	INTRODUCTION.	1-1
	1.1 Introduction	1-1
2	HYBRID VEHICLE DESIGN PROGRAM (HYVELD).	2-1
	2.1 HYVELD Description	2-1
	2.2 HYVELD Listing	2-5
3	HYBRID VEHICLE SIMULATION PROGRAM (HYVEC)	3-1
	3.1 HYVEC Description.	3-1
	3.2 HYVEC Listing.	3-5

LIST OF ILLUSTRATIONS

Figure		
2.1-1	Schematic of the Hybrid Vehicle Design (HYVELD) Calculation Procedure	2-2
3.1-1	Schematic of the Hybrid Vehicle Simulation Calculation (HYVEC)	3-3

Section 1

INTRODUCTION

1.1 INTRODUCTION

This is Volume III, Computer Program Listings, of Appendix B, Design Trade-Off Studies Report (Deliverable Item 2). It reports on work done on Task 2 and is part of Deliverable Item 7, Final Report, which is the summary report of a series which document the results of Phase I of the Near-Term Hybrid Vehicle Program. This phase of the program was a study leading to the preliminary design of a five-passenger hybrid vehicle utilizing two energy sources (electricity and gasoline/diesel fuel) to minimize petroleum usage on a fleet basis.

The program is sponsored by the U.S. Department of Energy (DOE) and the California Institute of Technology, Jet Propulsion Laboratory (JPL). Responsibility for this program at DOE resides in the Office of Electric and Hybrid Vehicle Systems. Work on this Phase I portion of the program was done by General Electric Corporate Research and Development and its subcontractors under JPL Contract 955190.

This volume contains a description and listing of two computer programs:

- Hybrid Vehicle Design Program (HYVELD)
- Hybrid Vehicle Simulation Program (HYVEC)

Both of the programs are modifications and extensions of similar programs developed at JPL as part of the Electric and Hybrid Vehicle System Research and Development Project, Hybrid Vehicle Potential Assessment, and Hybrid Vehicle System Evaluation Tasks.

Section 2

HYBRID VEHICLE DESIGN PROGRAM (HYVELD)

Section 2

HYBRID VEHICLE DESIGN PROGRAM (HYVELD)

2.1 HYVELD DESCRIPTION

This section contains a description and listing of the HYVELD computer program that was used during the Phase I Contract to study various vehicle designs.

The computer program (HYVELD) was developed as part of the Design Trade-Off Studies (Task 2) effort. It was used extensively to perform the first step in the screening of the various power train configurations and component combinations. In addition, it was used as the primary tool in the Sensitivity Analysis Studies (Task 4). A complete listing of the program is given in Section 2.2.

As indicated in Figure 2.1-1, the HYVELD calculation procedure consists of three parts: (1) Vehicle Synthesis, (2) Economics, (3) Energy-Use Comparisons. In the Vehicle Synthesis part of the program, the vehicle weight and cost, and the size and cost of the various power train components, are calculated for specified power train configurations and component characteristics. The passenger carrying capacity of the vehicle is set by inputting the appropriate baseline chassis weight, and the use-pattern is specified in terms of annual miles traveled and the fraction of those miles in urban driving. The vehicle performance is given in terms of power-to-weight ratio and electric range. Vehicle synthesis calculations are done sequentially for all-electric, series hybrids and parallel hybrids with and without secondary energy storage. Calculations are done for a single engine type and a number of battery types (e.g., lead-acid, Ni-Zn, Ni-Fe, Li-S) in each run.

The vehicle weight and cost for each power train configuration and component combination is built up from the Reference ICE Vehicle by subtracting the weight and cost of the conventional power train and adding the weight and cost of the hybrid/electric driveline needed to meet the specified vehicle performance. The effect on the vehicle weight of the added power train weight is accounted for by using a weight propagation factor.

Economics calculations are made for each of the power train combinations treated in the Vehicle Synthesis section of HYVELD. The objectives of the economics calculations are to determine the ownership cost (¢/mi), breakeven gasoline price ($/gal), and net dollars saved or lost ($/yr) for specified unit energy costs, economic conditions (interest, inflation, and discount rates), vehicle life, and maintenance costs (¢/mi). The Reference ICE Vehicle is characterized in terms of its initial cost, fuel economy, life, and maintenance costs. The ownership cost (¢/mi) of the Reference ICE Vehicle is calculated for comparison with that of the hybrid/electric vehicles.

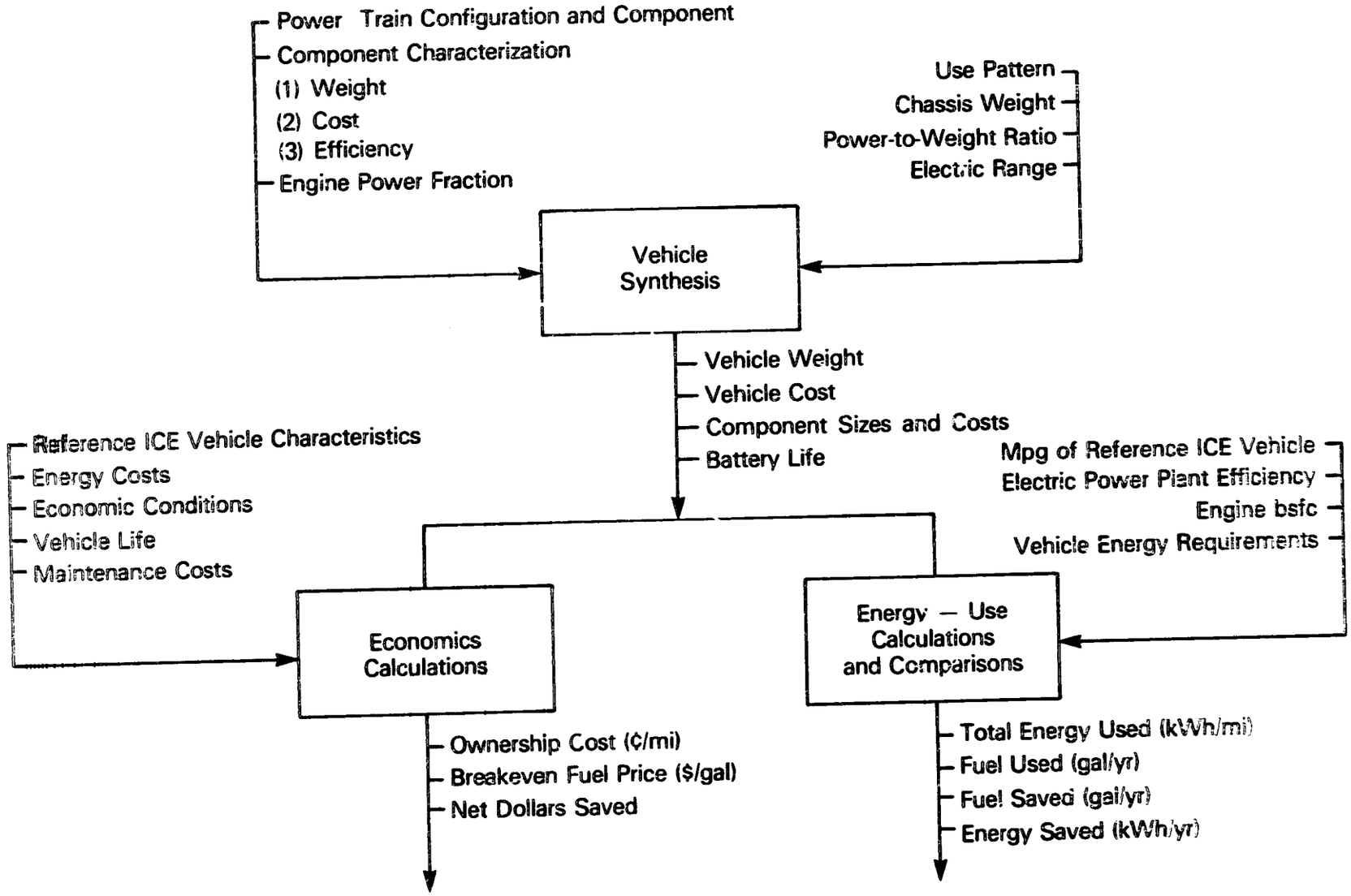


Figure 2.1-1. Schematic of the Hybrid Vehicle Design (HYVELD) Calculation Procedure

Energy-use calculations are also made for each of the power train combinations. Energy use (electricity and fuel) is calculated separately for urban and highway driving. The results are expressed both in terms of energy used per mile traveled and energy used per year. The fuel and energy used by the Reference ICE Vehicle is also calculated and compared with corresponding values for the hybrid/electric vehicles. Fuel and energy savings are then determined for each power train combination.

2.2 HYVELD LISTING

```
10C
20C***** ALL WRITE STATEMENTS REFER TO LU #16 INSTEAD
30C***** OF LU #6
40C
50C***** ALL READ STATEMENTS REFER TO LU #15 INSTEAD
60C***** OF LU #5
70C
80C     STORAGE SYSTEM
90C     SEX MUST BE NONZERO
100       REAL KPC,DTMG,CDTO
110       INTEGER IPTO,NSY,NBC,IPSU,NYL,NSYS,NBCS
120       INTEGER ECON
130       REAL IF,LRC,ICF,MCMO,MIFHV,NLDF,NSCV
140       REAL MPGU,MPGCB
150       REAL ROEPS,ROSPP,ROPPS,ROEPSL,ROPLL,ROES,RHOSS,BDPKWH,CP,CL,YLI
160       REAL KP,MPGH,K,YM,DP,RI
170       REAL DR,IR,TX,KPR,SWCNT,SWMG,SWCVT,SPCCVT
180       INTEGER TYPE,SYS,TYPESE,SYSSE
190       DIMENSION GAMMA(6),FDL(6),ALPHA(6)
200       DIMENSION ROEPS(5),ROSPP(5),ROPPS(5),ROEPSL(5),ROPLL(5),ROES(6)
210       DIMENSION RHOSS(5),BDPKWH(6),CP(6),CL(5),YLI(6),EE(5),BSV(5)
220       DIMENSION RI(6),FHE(6),FSS(6)
230       DIMENSION MIFHV(6)
240       NAMELIST /IN/SYS,TYPE,KP,ETAD,WI,K,CW,PW,KPC,
250      & SPCHE,SPCTRN,TMP,SEXU,SEXH,EFPHI,EFPUI,FUI,BSFCU,BSFCH,
260      & SPCMG,SPCGEN,SPCCNT,SPCTRN,DPKWHE,ETAM,HBFI,SWENG,NYL,
270      & SWTRN,AMC,GPO,MPGU,MPGCB,CHEF,MPGH,IPTO,NSY,NBC,IPSU,
280      & ROEPS,ROSPP,ROPPS,ROEPSL,ROPLL,ROES,RHOSS,BDPKWH,CP,CL,YLI,EE,
290      & FHE,FDL,FSS,YMI,DP,NSYS,NBCS,TYPESE,SYSSE,RI,GF,GK,CM,
300      & DR,IR,NS,CF,TX,KPR,SWCVT,SPCCVT,SWCNT,SWMG,SWGEN
310      & ,FMEDI,ETAPP,ECON,IF,NSCV,VICF,NLDF,ST,DMUP,CCV
320      & ,MIFHV,LRC,ICF,MCMO,FINC,UBCC,BSV
330C
340C****  COMMON BLOCKS ARE SET UP FOR USE IN THE SUBROUTINES
350C****  SEHYPR, CONVP, PRIMRY, PSECDY, ENCAL
360C****  COMMON  "INITV"  REFERS TO VARIABLES FROM INPUT DEVICE
370C****  COMMON  "INITM"  REFERS TO VARIABLES INITIALIZED IN MAIN
380C****  COMMON  "INITS"  REFERS TO VARIABLES INITIALIZED IN SUBROUTINES
390C
400       COMMON /INITV/SYS,TYPE,KP,ETAD,WI,K,CW,PW,KPC,
410      & SPCHE,TMP,SEXU,SEXH,EFPHI,EFPUI,FUI,BSFCU,BSFCH,
420      & SPCMG,SPCGEN,SPCCNT,SPCTRN,DPKWHE,ETAM,HBFI,SWENG,NYL,
430      & SWTRN,AMC,GPO,MPGU,MPGCB,CHEF,MPGH,IPTO,NSY,NBC,IPSU,
440      & ROEPS,ROSPP,ROPPS,ROEPSL,ROPLL,ROES,RHOSS,BDPKWH,CP,CL,YLI,EE,
450      & FHE,FDL,FSS,YMI,DP,NSYS,NBCS,TYPESE,SYSSE,RI,GF,GK,CM,
460      & DR,IR,NS,CF,TX,KPR,SWCVT,SPCCVT,SWCNT,SWMG,SWGEN
470      & ,FMEDI,ETAPP,ECON,IF,NSCV,VICF,NLDF,ST,DMUP,CCV
480      & ,MIFHV,LRC,ICF,MCMO,FINC,UBCC,BSV
490C
500       COMMON /INITM/SEX,WVO,ETAEU,ETAEP,WDTO,GAMMA,YMP,ALPHA
510       COMMON /INITS/CDTO,DTMG,SEXHCV,OPCTG,TACCV,AGCCV,EFPU,EFPH,FU,YM
520C
530C END OF REAL/INTEGER/DIMENSION/COMMON STUFF
540C
550C
560C     SYS=1  PURE STORAGE=PRIMARY
570C     SYS=2  PURE STORAGE=PRIMARY+SECONDARY
580C     SYS=3  SERIES HYBRID PRIMARY=SECONDARY
590C     SYS=4  PARALLEL HYBRID PRIMARY ONLY
600C     SYS=5  PARALLEL HYBRID PRIMARY=SECONDARY
```

```
610C     SYS=6  SERIES HYBRID-PRIMARY ONLY
620C  RID=RANGE OF ALL ELECTRIC VEH  (=RI(1))
630C  EE=ENERGY EXPONENT
640C
650C NSCV = LIFETIM OF CONVENTIONAL VEHICLE
660C CCV  = COST OF CONVENTIONAL VEHICLE ($)
670C VICF = VEHICLE IMPROVEMENT COST FACTOR
680C NLDF = DEPRECIATION FACTOR
690C ST   = SALES TAX
700C DMUP = DEALER MARKUP
710C UBCC = BATTERY CHARGER COST ($/KWH)
720C
730       CALL FILES
740       READ(15,1N)
750       IF(IPTO.EQ.0)GO TO 500
760       IF(IPTO.EQ.1)GO TO 100
770C
780C ********  D E S I G N   P A R A M E T E R S ************
790C
800   100 WRITE(16,110)
810   110 FORMAT(2X,'TABLE A:DESIGN PARAMETERS FOR DIFF. SIZE VEHICLES')
820       WRITE(16,120)
830   120 FORMAT(20X,'VEHICLE TYPE')
840       WRITE(16,130)
850   130 FORMAT(5X,'DESIGN PARAMETER',30X,'DIFF. SIZE')
860       WRITE(16,140)CW
870   140 FORMAT(2X,'CHASSIS WEIGHT(LBS)',T50,F10.3)
880       WRITE(16,150)PW
890   150 FORMAT(2X,'PAYLOAD WEIGHT(LBS)',T50,F10.3)
900   160 FORMAT(2X,'POWER TO WEIGHT RATION(ELECTRIC DRIVE)',T50,F10.3)
910       WRITE(16,170)KP
920   170 FORMAT(2X,'POWER TO WEIGHT RATIO',T50,
930      & F10.3)
940       WRITE(16,180)W1
950   180 FORMAT(2X,'BASE LOADED VEHICLE(LBS)',T50,F10.3)
960   190 FORMAT(2X,'POWERTRAIN WEIGHT(LBS):CONVENTIONAL VEHICLE',T50,
970      & F10.3)
980       WRITE(16,200)RI(1)
990   200 FORMAT(2X,'RANGE(DESIGN)-MILES:PURE STORAGE',T50,F10.3)
1000       WRITE(16,210)RI(3)
1010  210 FORMAT(2X,'RANGE(DESIGN)-MILES:HYBRID CITY DRIVING',T50,F10.3)
1020       WRITE(16,220)ETAD
1030  220 FORMAT(2X,'ELECTRIC DRIVELINE EFFICIENCY',T50,F10.3)
1040       WRITE(16,225) ETAM
1050  225 FORMAT(2X,'MECHANICAL DRIVE EFFICIENCY',T50,F10.3)
1060       WRITE(16,230) ETAPP
1070  230 FORMAT(2X,'POWER PLANT EFFECIENCY',T50,F10.3)
1080C     NSY= SYSTEM CODE
1090C     NSY= 6  ALL6SYSTEMS
1100C     NBC= 5  ALL5BATTERY CODE
1110C     IPTO= 0  RESULT ONLY
1120C     IPTO= 1  RESULTS + VEHICLE DESIGN PARAMETER
1130C     NYL = 0  IF INPUT BATTERY YEARS TO BE USED
1140C     NYL = 1  IF BATTERY LIFE IS TO BE CALCULATED FOR CYCLES
1150C     IPSU= 0  STORAGE UNIT TABLE AS USED IN CALCULATION (NO TABLE)
1160C     IPSU= 1  PRINT TABLE
1170       WRITE(16,240)AWC
```

```
1070  230 FORMAT(2X,'POWER PLANT EFFECIENCY',T50,F10.3)
1080C     NSY= SYSTEM CODE
1090C     NSY= 6  ALL6SYSTEMS
1100C     NBC= 5  ALL5BATTERY CODE
1110C     IPTO= 0  RESULT ONLY
1120C     IPTO= 1  RESULTS + VEHICLE DESIGN PARAMETER
1130C     NYL = 0  IF INPUT BATTERY YEARS TO BE USED
1140C     NYL = 1  IF BATTERY LIFE IS TO BE CALCULATED FOR CYCLES
1150C     IPSU= 0  STORAGE UNIT TABLE AS USED IN CALCULATION (NO TABLE)
1160C     IPSU= 1  PRINT TABLE
1170       WRITE(16,240)AWC
1180  240 FORMAT(2X,'COST OF ADDITIONAL CHASSIS WEIGHT($/LB)',T50,F10.3)
1190       WRITE(16,250)K
1200  250 FORMAT(2X,'WEIGHT PROPAGATION FACTOR',T50,F10.3)



1210       WRITE(16,260)YMI
1220  260 FORMAT(2X,'MILES TRAVELED PER YEAR',T50,F10.3)
1230       WRITE(16,270)FUI
1240  270 FORMAT(2X,'FRACTION OF MILES IN CITY',T50,F10.3)
1250       WRITE(16,280) FMEDI
1260  280 FORMAT(2X,'FRACTION OF MILES IN CITY, ELECTRIC',T50,F10.3)
1270       WRITE(16,290)SEXU
1280  290 FORMAT(2X,'ENERGY CONSUMPTION IN CITY(KWH/TON-MILE)',T50,F10.3)
1290       WRITE(16,300)SEXH
1300  300 FORMAT(2X,'ENERGY CONSUMPTION ON HIGHWAY(KWH/TON-MILE)',T50,F10.3)
1310       WRITE(16,310)EFPUI
1320  310 FORMAT(2X,'FRACTION OF ENERGY FROM ENGINE IN CITY',T50,F10.3)
1330       WRITE(16,320)EFPHI
1340  320 FORMAT(2X,'FRACTION OF ENERGY FROM ENGINE ON HIGHWAY',T50,F10.3)
1350       WRITE(16,330)DPKWHE
1360  330 FORMAT(2X,'PRICE OF ELECTRICITY($/KWH)',T50,F10.3)
1370       WRITE(16,340) GPO
1380  340 FORMAT(2X,'GASOLINE PRICE ($/GAL)',T50,F10.3)
1390       WRITE(16,341)SWMG
1400  341 FORMAT(2X,'SPECIFIC WEIGHT OF MOTOR/GENERATOR (LB/KW)',T50,F10.3)
1410       WRITE(16,342)SWGEN
1420  342 FORMAT(2X,'SPECIFIC WEIGHT OF GENERATOR (LB/KW)',T50,F10.3)
1430       WRITE(16,343)SWCNT
1440  343 FORMAT(2X,'SPECIFIC WEIGHT OF CONTROLLER (LB/KW)',T50,F10.3)
1450       WRITE(16,344)UBCC
1460  344 FORMAT(2X,'SPECIFIC COST OF BATTERY CHARGER ($/KWH)',T50,F10.3)
1470       WRITE(16,350)SPCMG
1480  350 FORMAT(2X,'SPECIFIC COST OF MOTOR/GENERATOR',T50,F10.3)
1490       WRITE(16,360)SPCGEN
1500  360 FORMAT(2X,'SPECIFIC COST OF GENERATOR($/KW)',T50,F10.3)
1510       WRITE(16,370)SPCCNT
1520  370 FORMAT(2X,'SPECIFIC COST OF CONTROLLER($/KW)',T50,F10.3)
1530       WRITE(16,380)HBFI
1540  380 FORMAT(2X,'HYBRID BATTERY FACTOR(HIGHWAY)',T50,F10.3)
1550       WRITE(16,390)CHEF
1560  390 FORMAT(2X,'BATTERY CHARGING EFFICIENCY FROM PLUG',T50,F10.3)
1570       WRITE(16,400)BSFCU
1580  400 FORMAT(2X,'ENGINE BSFC IN CITY(LB/BHP-HR)',T50,F10.3)
1590       WRITE(16,410)BSFCH
1600  410 FORMAT(2X,'ENGINE BSFC ON HIGHWAY(LB/BHP-HR)',T50,F10.3)
1610       WRITE(16,420)TMP
1620  420 FORMAT(2X,'TIME FOR SUSTAINED POWER FROM F.W.(SEC)',T50,F10.3)
1630       WRITE(16,421) VICF
1640  421 FORMAT(2X,'VEHICLE IMPROVEMENT COST FACTOR',T50,F10.3)
1650       WRITE(16,422) (I,MIFHV(I),I=1,NSY)
1660  422 FORMAT(2X,'MAINTENCE IMPROVEMENT FACTOR: SYSTEM=',I1,T50,F10.3)
```

```
1670C
1680C ******** E C O N O M I C   F A C T O R S **********
1690C
1700       WRITE(16,430)
1710  430 FORMAT(//,2X,'TABLE B:  ECONOMIC FACTORS',/)
1720       WRITE(16,440) DR
1730  440 FORMAT(2X,'DISCOUNT RATE',T50,F10.3)
1740       WRITE(16,450) IR
1750  450 FORMAT(2X,'INTEREST RATE',T50,F10.3)
1760       WRITE(16,460) NS
1770  460 FORMAT(2X,'PAYBACK PERIOD STRUCTURE (YRS)',T50,I10)
1780       WRITE(16,470) NF
1790  470 FORMAT(2X,'FINANCE PERIOD (YRS)',T50,I10)
1800       WRITE(16,480) TX




1810  480 FORMAT(2X,'TAX RATE',T50,F10.3)
1820       WRITE(16,490) IF
1830  490 FORMAT(2X,'INFLATION FACTOR',T50,F10.3)
1840       WRITE(16,491) ST
1850  491 FORMAT(2X,'SALES TAX',T50,F10.3)
1860       WRITE(16,492) DMUP
1870  492 FORMAT(2X,'DEALER MARK UP',T50,F10.3)
1880       WRITE(16,493) NLDF
1890  493 FORMAT(2X,'NONLINEAR DEPRECIATION FACTOR',T50,F10.3)
1900  500 SEX=SEXU
1910       ETAEU=.134/BSFCU
1920       ETAEH=.134/BSFCH
1930       ALPHAC=SWENG+SWTRN
1940       WVO=W1/(1.-(ALPHAC*KPC))
1950       WDTO=WVO*KPC*ALPHAC
1960       YMP=YMI
1970       IF(IPTO.EQ.0)GO TO 980
1980C
1990C********** CONVENTIONAL VEHICLE DESCRIPTION ***********
2000C
2010       WRITE(16,510)
2020  510 FORMAT(//,2X,'CONVENTIONAL VEHICLE')
2030       WRITE(16,520)KPC
2040  520 FORMAT(2X,'POWER TO WEIGHT RATIO(KW/LB)',T50,F10.3)
2050  530 FORMAT(2X,'WEIGHT POWERTRAIN(LBS)',T50,F10.3)
2060       WRITE(16,540) SWENG
2070  540 FORMAT(2X,'SPECIFIC WEIGHT ENGINE(LB/KW)',T50,F10.3)
2080       WRITE(16,550)SWTRN
2090  550 FORMAT(2X,'SPECIFIC WEIGHT TRANSMISSION(LB/KW)',T50,F10.3)
2100       WRITE(16,560)SPCHE
2110  560 FORMAT(2X,'SPECIFIC COST ENGINE($/KW)',T50,F10.3)
2120       WRITE(16,570)SPCTRN
2130  570 FORMAT(2X,'SPECIFIC COST TRANSMISSION($/KW)',T50,F10.3)
2140       WRITE(16,580)MPGU
2150  580 FORMAT(2X,'FUEL ECONOMY IN CITY(MILES/GAL)',T50,F10.3)
2160       WRITE(16,590)MPGH
2170  590 FORMAT(2X,'FUEL ECONOMY ON HIGHWAY(MILES/GAL)',T50,F10.3)
2180       WRITE(16,
```

```
      WRITE(16,600)MPGCB
600   FORMAT(2X,'FUEL ECONOMY(COMPOSITE)',T50,F10.3)
      WRITE(16,601) CCV
601   FORMAT(2X,'CONVENTIONAL VEHICLE COST ($)',T50,F10.3)
      WRITE(16,602) NSCV
602   FORMAT(2X,'CONVENTIONAL VEHICLE LIFETIME (YR)',T50,F10.3)
C
C********* SUMMARY OF CONFIGURATIONS **********
C
      WRITE(16,610)
610   FORMAT(//,2X,'SUMMARY OF DRIVELINE CONFIGURATIONS EVALUATED')
      WRITE(16,620)
620   FORMAT(2X,'1-PURE STORAGE VEHICLES',T50,'PRIMARY STORAGE',
     & T70,'SECONDARY STORAGE',T89,'FSS')
      WRITE(16,630)
630   FORMAT(2X,'DRIVELINE CONFIGURATION')
      WRITE(16,640)
640   FORMAT(2X,'ELECTRIC:PRIMARY',T50,'PB ACID A')
      WRITE(16,650)
650   FORMAT(T50,'NI-ZN')
      WRITE(16,660)
660   FORMAT(T50,'NI-FE')
      WRITE(16,670)
670   FORMAT(T50,'LI-S')
C
C********* ELECTRIC VEHICLE DESCRIPTION **********
C
      WRITE(16,680)FSS(2)
680   FORMAT(2X,'ELECTRIC:PPRIMARY/SECONDARY',T50,'LEAD ACID',T70,'LEAD
     & ACID',T89,F4.2)
      WRITE(16,690)FSS(2)
690   FORMAT(T50,'LEAD ACID',T70,'FLYWHEEL',T89,F4.2)
      WRITE(16,700)FSS(2)
700   FORMAT(T50,'LI-S',T70,'LEAD ACID',T89,F4.2)
      WRITE(16,710)FSS(2)
710   FORMAT(T50,'LI-S',T70,'FLYWHEEL',T89,F4.2)
C
C********** HYBRID VEHICLE DESCRIPTION ************
C
      WRITE(16,720)
720   FORMAT(2X,'HYBRID(HEAT ENGINE/ELECTRIC)VEHICLES',T50,
     & 'PRIMARY STORAGE',T70,'SECONDARY STORAGE',T89,'FSS',T94,'FDE',
     & T100,'FHE')
      WRITE(16,730)
730   FORMAT(2X,'SERIES(PRIMARY)',T50,'LEAD ACID',T94,'1.0',T100,'.33')
      WRITE(16,740)FSS(3),FDE(3),FHE(3)
740   FORMAT(2X,'SERIES(PRIMARY/SECONDARY)',T50,'LEAD ACID',T70,
     & 'LEAD ACID',T89,F4.2,T94,F3.1,T99,F4.2)
      WRITE(16,750)FSS(3),FDE(3),FHE(3)
750   FORMAT(T50,'LEAD ACID',T70,'FLYWHEEL',T89,F4.2,T94,F3.1,T99,
```

```
     & F4.2)
      WRITE(16,760)FSS(3),FDE(3),FHE(3)
760 FORMAT(T50,'LI-S',T70,'LEAD ACID',T89,F4.2,T94,F3.1,T99,F4.2)
      WRITE(16,770)FSS(3),FDE(3),FHE(3)
770 FORMAT(T50,'LI-S',T70,'FLYWHEEL',T89,F4.2,T94,F3.1,T99,F4.2)
      WRITE(16,780)FDE(4),FHE(4)
780 FORMAT(2X,'PARALLEL(PRIMARY)',T50,'PB ACID-A',T94,F3.1,T99,F4.2)
      WRITE(16,790)FDE(4),FHE(4)
790 FORMAT(T50,'NI-ZN',T94,F3.1,T99,F4.2)
      WRITE(16,800)FDE(4),FHE(4)
800 FORMAT(T50,'NI-FE',T94,F3.1,T99,F4.2)
      WRITE(16,810)FDE(4),FHE(4)
810 FORMAT(T50,'LI-S',T94,F3.1,T99,F4.2)
      WRITE(16,820)FSS(5),FDE(5),FHE(5)
820 FORMAT(2X,'PARALLEL(PRIMARY/SECONDARY)',T50,'LEAD ACID',T70,
     & 'LEAD ACID',T89,F4.2,T94,F3.1,T99,F4.2)
      WRITE(16,830)FSS(5),FDE(5),FHE(5)
830 FORMAT(T50,'LEAD ACID',T70,'FLYWHEEL',T89,F4.2,T94,F3.1,T99
     & ,F4.2)
      WRITE(16,840)FSS(5),FDE(5),FHE(5)
840 FORMAT(T50,'LI-S',T70,'LEAD ACID',T89,F4.2,T94,F3.1,T99,F4.2)
      WRITE(16,850)FSS(5),FDE(5),FHE(5)
850 FORMAT(T50,'LI-S',T70,'FLYWHEEL',T89,F4.2,T94,F3.1,T99,F4.2)
C
C******* T A B L E  C -- S T O R A G E  R E Q U I R E M E N T S ******
C
      WRITE(16,860)
860 FORMAT(2X,///)
      WRITE(16,870)
870 FORMAT(2X,'TABLE C:STORAGE UNIT CHARACTERISTICS USED IN THE DESIGN
     & CALCULATIONS')
      WRITE(16,880)
880 FORMAT(2X,T25,'PRIMARY STORAGE',T50,'SECONDARY STORAGE',T70,'COST'
```

```
     & T80,'CYCLE LIFE',T95,'LIFETIME',T104,'ENERGY',T112,'BATTERY')
      WRITE(16,890)
890 FORMAT(2X,T20,'WITHOUT',T40,'WITH',T95,'YEAR',T106,'EXP',
     & T112,'SALVAGE')
      WRITE(16,900)
900 FORMAT(2X,T20,'LOAD LEVELING',T40,'LOAD LEVELING')
      WRITE(16,910)
910 FORMAT(2X,'STORAGE UNIT',T20,'WH/LB',T26,'W/LB',T31,'WP/LP',
     & T41,'WH/LB',T47,'W/LB',T52,'WH/LB',T58,'W/LB',T65,'$/KWH',T72,
     & '$/LB',T80,'CYCLES',T95,'NO')
      WRITE(16,920)
920 FORMAT(2X,'BATTERY TYPE')
      WRITE(16,930)ROEPS(1),ROSPP(1),ROPPS(1),ROEPSL(1),ROPLL(1),
     & ROES(1),RHOSS(1),BDPKWH(1),CP(1),CL(1),YLI(1),EE(1),BSV(1)
930 FORMAT(5X,'LEAD ACID',T20,F4.0,T26,F5.0,T31,F5.0,T41,F5.0,T47,
     & F4.0,T52,F4.0,T58,F4.0,T65,F6.0,T72,F4.2,T80,F8.2,T95,F4.0,
     & T105,F4.2,T113,F4.2)
      WRITE(16,940)ROEPS(2),ROSPP(2),ROPPS(2),ROEPSL(2),ROPLL(2),
     & ROES(2),RHOSS(2),BDPKWH(2),CP(2),CL(2),YLI(2),EE(2),BSV(2)
940 FORMAT(5X,'LEAD ACID(ADV)',T20,F4.0,T26,F5.0,T31,F5.0,T41,F5.0,T47
     & ,F4.0,T52,F4.0,T58,F4.0,T65,F6.0,T72,F4.2,T80,F8.2,T95,F4.0,
     & T105,F4.2,T113,F4.2)
      WRITE(16,950)ROEPS(3),ROSPP(3),ROPPS(3),ROEPSL(3),ROPLL(3),
     & ROES(3),RHOSS(3),BDPKWH(3),CP(3),CL(3),YLI(3),EE(3),BSV(3)
```

```
3250  950 FORMAT(5X,'NI= 2N',T20,F4.0,T26,F5.0,T31,F5.0,T41,F5.0,T47,F4.0,
3260     & T52,F4.0,T58,F4.0,T65,F6.0,T72,F4.2,T80,F8.2,T95,F4.0,
3270     & T105,F4.2,T113,F4.2)
3280      WRITE(16,960)ROEPS(4),ROSPP(4),ROPPS(4),ROEPSL(4),ROPLL(4),
3290     & ROES(4),RHOSS(4),BDPKWH(4),CP(4),CL(4),YLI(4),EE(4),BSV(4)
3300  960 FORMAT(5X,'NI FL',T20,F4.0,T26,F5.0,T31,F5.0,T41,F5.0,T47,
3310     & F4.0,T52,F4.0,T58,F4.0,T65,F6.0,T72,F4.2,T80,F8.2,T95,F4.0,
3320     & T105,F4.2,T113,F4.2)
3330      WRITE(16,970)ROEPS(5),ROSPP(5),ROPPS(5),ROEPSL(5),ROPLL(5),
3340     & ROES(5),RHOSS(5),BDPKWH(5),CP(5),CL(5),YLI(5),EE(5),BSV(5)
3350  970 FORMAT(5X,'LI-S',T20,F4.0,T26,F5.0,T31,F5.0,T41,F5.0,T47,F4.0,T52,
3360     & F4.0,T58,F4.0,T65,F6.0,T72,F4.2,T80,F8.2,T95,F4.0,
3370     & T105,F4.2,T113,F4.2)
3380      WRITE(16,975) ROES(6),BDPKWH(6),CP(6),YLI(6)
3390  975 FORMAT(5X,'FLYWHEEL',T52,F4.0,T65,F6.0,T72,F4.2,T95,F4.0)
3400      WRITE(16,860)
3410C
3420C*****************************************************
3430C
3440C   M A I N   P R O G R A M   L O O P
3450C
3460C*****************************************************
3470C
3480  980 DO 20000 SYS=NSYS,NSY
3490C     SPECIFIC COST IN $/KW
3500       GAMMA(SYS)=(FIL(SYS)*SPCHE)+SPCTRN+(FDE(SYS)
3510     & *(1./CM*SPCMG+SPCCNT))
3520C     SPECIFIC WEIGHT IN LB/KW
3530       ALPHA(SYS)=(FIL(SYS)*SWENG)+SWTRN+(FDE(SYS)*(SWCNT+SWMG/CM))
3540       IF(SYS.EQ.1)CALL PUREP
3550       IF(SYS.EQ.2)CALL PSECDY(SYS)
3560       IF(SYS.EQ.3)CALL PSECDY(SYS)
3570       IF(SYS.EQ.4)CALL PARAHP
3580       IF(SYS.EQ.5)CALL PSECDY(SYS)
3590       IF (SYS.EQ.6)CALL SEHYPR
3600       WRITE(16,990)




3610  990 FORMAT(1H0)
3620*20000 CONTINUE
3630C
3640C CLOSE FILES   DETACH FCS FROM AFT
3650C
3660       CALL FILED
3670       STOP
3680       END
3690       SUBROUTINE SEHYPR
3700C
3710C*****************************************************
3720C  S U B R O U T I N E        S E H Y P R
3730C*****************************************************
3740C
3750       REAL  DPC,DTMG,CDTO
```

```
3750        REAL KPC,DTMG,CDTO
3760        INTEGER IPTO,NSY,NBC,IPSU,NYL,NSYS,NBCS
3770        INTEGER ECON
3780        REAL MPGU,MPGCH
3790        REAL ROEPS,ROSPP,ROPPS,ROEPSL,ROPLL,ROES,RHOSS,BDPKWH,CP,CL,YLI
3800        REAL KP,MPGH,K,DP,RI
3810        REAL DR,IR,TX,KPR,SWCNT,SWMG,SWCVT,SPCCVT
3820        REAL IRE
3830        REAL IF,LRC,ICF,MCMO,MIFHV,NLDF,NSCV
3840        INTEGER TYPE,SYS,TYPESE,SYSSE
3850        DIMENSION GAMMA(6),FDE(6),ALPHA(6)
3860        DIMENSION ROEPS(5),ROSPP(5),ROPPS(5),ROEPSL(5),ROPLL(5),ROES(6)
3870        DIMENSION RHOSS(5),BDPKWH(6),CP(6),CL(5),YLI(6),EE(5),BSV(5)
3880        DIMENSION RI(6),FHE(6),FSS(6)
3890        DIMENSION MIFHV(6)
3900        COMMON /INITV/SYS,TYPE,KP,ETAD,W1,K,CW,PW,KPC,
3910       & SPCHE,TMP,SEXU,SEXH,EFPHI,EFPUI,FUI,BSFCU,BSFCH,
3920       & SPCMG,SPCGEN,SPCCNT,SPCTRN,DPKWHE,ETAM,HBFI,SWENG,NYL,
3930       & SWTRN,AWC,GPO,MPGU,MPGCB,CHEF,MPGH,IPTO,NSY,NBC,IPSU,
3940       & ROEPS,ROSPP,ROPPS,ROEPSL,ROPLL,ROES,RHOSS,BDPKWH,CP,CL,YLI,EE,
3950       & FHE,FDE,FSS,YMI,DP,NSYS,NBCS,TYPESE,SYSSE,RI,GF,GK,CM,
3960       & DR,IR,NS,NF,TX,KPR,SWCVT,SPCCVT,SWCNT,SWMG,SWGEN
3970       & ,FMEDI,ETAPP,ECON,IF,NSCV,VICF,NLDF,ST,DMUP,CCV
3980       & ,MIFHV,LRC,ICF,MCMO,FINC,UBCC,BSV
3990C
4000        COMMON /INITM/SEX,WVO,ETAEU,ETAEH,WDTO,GAMMA,YMP,ALPHA
4010        COMMON /INITS/CDTO,DTMG,SEXHCV,OPCTG,TACCV,AGCCV,EFPU,EFPH,FU,YM
4020C
4030        REAL NBEUPG
4040        REAL MPG
4050        REAL IC
4060C
4070C END OF REAL/INTEGER/DIMENSION/COMMON STUFF
4080C
4090        EFPU=EFPUI
4100        EFPH=EFPHI
4110        HBF=HBFI
4120        FU=FUI
4130        YM=YMI
4140        FMED=FMEDI
4150        TYPE=TYPESE
4160        SYS=SYSSE
4           FOR SERIES HYBRID PRIMARY ONLY - LEAD ACID TYPE=1
4180C       RI(6)=30.
4190        RIS=RI(SYS)
4200        RIO=RI(1)
```

```
      ALPHAS=ALPHA(SYS)+(FHE(SYS)*SWGEN)
      ROPS1=ROPS(TYPE)*( RIS/RIO )**FE(TYPE)
      PHISPP=SQRT(2./ETAD/ROPS1
      PHIPP=(1.-FHE(SYS))*1000.*KP/ROPPS(TYPE)
      PHISPP=1000.*(GK=  (GF*KP*FHE(SYS)))/ROSPP(TYPE)
      IF(PHISPP.GT.PHIPP)PHI=PHIPP
      IF(PHISPP.GT.PHIPP)PHI=PHIPP
      IF(PHISPP.GT.PHI)PHI=PHISPP
      RR=RIO*( 2.*ETAD*ROPS(TYPE)*PHI/(SEX*RIO) )**
     & (1./(1.-FE(TYPE)) )
      WV1R=1./(1.-((ALPHAS*KP))-((1.+K)*PHI))
      WV=WV1R*W1
      WDT1R=WV1R-1.
      WDT=WDT1R*W1
      WB=WV*PHI
      WRITE(16,100)
  100 FORMAT(//,2X,'SERIES HYBRID:PRIMARY ONLY',5X,'PHI',7X,'RR',
     & 8X,'WV1R',5X,'WV',8X,'WDT1R',5X,'WDT',5X,'WB',4X,'YL')
      YLS=YL1(TYPE)
      IF(NYL.EQ.0)YL=YLS
      IF(NYL.EQ.1)YL=2.*DP*CL(TYPE)*ROEPS(TYPE)*PHI*ETAD/SEX/YM/HBF/FU
C     HBF=1. FOR PRIMARY (HYBRID BATTERY FACTOR)
      WRITE(16,110)PHI,RR,WV1R,WV,WDT1R,WDT,WB,YL
  110 FORMAT(5X,'BATTERY TYPE:LEAD ACID',F10.3,F10.0,F10.2,F10.0,
     & F10.2,2F10.0,F7.2)
C
C HEADER FOR DRIVELINE COST
C
      WRITE(16,120)
  120 FORMAT(//,2X,'DRIVELINE COST')
      WRITE(16,140)
  140 FORMAT(2X,'PRIMARY',T17,'PRIMARY',T30,'SECONDARY'
     & ,T45,'SECONDARY',T65,'TOTAL',T84,'INCREMENTAL',T100,'EHV',/,4X
     & ,'TYPE',T19,'COST',T34,'TYPE',T48,'COST',T60,'DRIVELINE COST'
     & ,T82,'DRIVELINE COST',T100,'COST')
C     PRIMARY COST
      PC=WV*PHI*CP(TYPE)*(1.+DMUP)
      CBCH=(WV*PHI*ROEPS1*UBCC*(1.+DMUP)) / 1000.
C     TOTAL DRIVETRAIN COST CDT
      GAMMAS=GAMMA(SYS)+(FHE(SYS)*SPCGEN)
      CDT=PC+(WV*KP*GAMMAS)*(1.+DMUP) + CBCH
C     CONVENTIONAL COST CDTO(HEAT ENGINE+TRN)
      CDTC=WVO*KP*(SPCHE+SPCTRN)*(1.+DMUP)
C     THE INCREMENTAL DRIVELINE COST CDTI
      CDTI=CDT-CDTO
C     COST OF ADDITIONAL WEIGHT OF CAR AT (AWC)$/LB
      DELCW=K*PHI*WV*(1.+DMUP)
      IC=CDTI-PC*(1.-BSV(TYPE))
      DELCDT=IC+DELCW
C     CPMEOP=CENT PER MILE OF EQUIPMENT
C
C EVALUATE CHEVS & NLDF (IF NLDF NOT EQUAL TO 1 BY USER)
C
      IF(NLDF.NE.1.) NLDF= (2.*NS-NF+1.)/(NS+1.)
      CHEVS=DELCDT + (CCV-CDTO)*(1.+VICF)
      CEHV=CHEVS + PC*(1.-BSV(TYPE))
      IRE=(1.-TX)*IR
      FF=NLF*IRE/(1.-(1.+IRE)**(-NF))
      FF0=FF* (DR-IF)/(1.+IF)
      FF1=(1.+DR)/(1.+IF)
```

```
      FRCS=FF0 / (1.-FF1**(-NS))
      FRCU=FF0 / (1.-FF1**(-YL))
      CPMEOP=100./YM*((FRCS*NLDF*CHEVS) +
     & (FRCU*(1.-BSV(TYPE))*WV*PHI*CP(TYPE)))
C     DPTMEQ=DOLLARS PER TON MILES EQUIPMENT
      DPTMEQ=CPMEOP/100./(W1/2000.)
C     DPTMEL=DOLLARS PER TON MILES ELECTRIC
C     DPKWHE=DOLLARS PER KILOWATT HOUR ELECTRIC
      IF(FU.EQ.1.)MPG=MPGU
      IF(FU.NE.1.)MPG=MPGCB
      SEXHCV=35.1/MPG/WVO*2000.
      DTMG=GPO/MPG/W1*2000.
      DPTMEL=(WV/W1)*DPKWHE*((FU*FMED*SEXU*(1.-EFPU)/ETAD)+((1.-FU)
     & *SEXH*(1.-EFPH)/ETAD))/CHEF
C
      FRCV= FF0 / (1. - FF1**(-NSCV))
      DFCV=(2.*NSCV-NF+1.) / (NSCV+1.)
      DTMCV=(FRCV*DFCV*CCV*(1.+ST)) / (YM*W1/2000.)
C
C LRC  = LICENSE AND REGISTER FEE
C FINC = FIXED INSURANCE COST
C ICF  = INSURANCE COST FACTOR
C MCMO = MAINTINENCE COST PER MILE FOR CV
C MIFHV = MAINTINANCE IMPROVEMENT FACTOR FOR HYBRID VEHICLE
C
C
      BC=PC
      ADCHV=LRC+FINC+ICF*(CHEVS+BC)
      ADCCV=LRC+FINC+ICF*CCV
      MCHV=.01*MCMO*YM*(1.-MIFHV(SYS))
      MCCV=.01*MCMO*YM
C
      NBEUPG=DPTMEQ+DPTMEL
     & - DTMCV + (ADCHV-ADCCV+MCHV-MCCV) / (YM*W1/2000.)
      DBEUPG=.03 * ( SEXHCV*WVO/W1 -
     & WV1R*( FU*FMED*SEXU*EFPU / (ETAEU*ETAM) +
     &        (1.-FU)*SEXH*EFPH / (ETAEH*ETAM) +
     &   FU*(1.-FMED)*SEXU / (ETAM*ETAEU) ) )
C     BEUPG=BREAK EVEN UNIT PRICE OF GASOLINE
      BEUPG=NBEUPG/DBEUPG
      GCPMHV=0.
      IF (FU.EQ.1.) GOTO 145
      GCPMHV=GPO*(WV/2000.)/(36.63*ETAM) *
     & ( SEXU*FU*FMED*EFPU / ETAEU + SEXU*FU*(1.-FMED) / ETAEU +
     &   SEXH*(1.-FU)*EFPH / ETAEH )
  145 CONTINUE
C
C  CALCULATE OPERATING COST PER MILE (CENTS/MI)
C
```

```
      OPCTEHV=100.*( (W1/2000.)*(DPTMEO+DPTMEL) +
     &                (ADCHV+MCHV)/YM + GCPMHV )
      OPCTG=100.*( (W1/2000.)*(DTMCV+DTMG) + (ADCCV+MCCV)/YM )
C
C CALCULATE TOTAL ANNUAL OPERATING COST ($/YR)  TACEHV
C
      AOPCG=GCPMHV*YM
      TACEHV=OPCTEHV*(YM/100.)
      AOPCEQE=TACEHV-AOPCG-ADCHV-MCHV
      TACCV=OPCTG*(YM/100.)
      AGCCV=DTMG*(W1/2000.)*YM




      WRITE(16,150)PC,CDT,CDTI,CEHV
  150 FORMAT(2X,'PB-ACID',T17,F10.0,T60,F10.0,T82,F10.0,T95,F10.0)
C
C HEADER FOR BREAKEVEN GAS PRICES
C
      WRITE(16,160)
  160 FORMAT(//,2X,'OPERATING COSTS AND BREAK EVEN GASOLINE PRICES')
      WRITE(16,170)
  170 FORMAT(T59,'GAS BKEV',T70,'OPER COST EQ&EL',T87,'HV GAS-PO'
     & ,T103,'DRIVE CHAR')
      WRITE(16,180)
  180 FORMAT(2X,'TYPE',T19,'DELCDT',T27,'BATTERY COST',T40,'DPTMEQ'
     & ,T50,'DPTEML',T57,'GAS $/GAL',T67,'OPER $/MI',T82,'$/YR',T92,
     & '$/YR',T101,'LB/KW',T112,'$/KW')
      WRITE(16,190) DELCDT,PC,DPTMEQ,DPTMEL,BEUPG,
     & OPCTEHV,AOPCEQE,AOPCG,ALPHAS,GAMMAS
  190 FORMAT(2X,'LEAD ACID',T15,F10.0,T26,F10.0,T37,F9.3,T47,F9.3
     & ,T57,F9.2,T67,F9.2,T77,F9.2,T87,F9.2,T97,F9.2,T107,F9.2)
C
C OUTPUT ENERGY CALCULATIONS IF ECON=1
C
      NTYPE=1
      IF (ECON.EQ.1) CALL ENCAL(WV,NTYPE)
C
C OUTPUT CONVENTIONAL VEHICLE OUTPUT
C
      CALL CONVP
      RETURN
      END
      SUBROUTINE CONVP
C
C*********************************************
C  S U B R O U T I N E        C O N V P
C*********************************************
C
C    CONVENTIONAL VEHICLE PRINT
      REAL KPC,DTMG,CDTO
      REAL IF,LRC,ICF,MCMO,MIFHV,NLDF,NSCV
      INTEGER IPTO,NSY,NBC,IPSU,NYL,NSYS,NBCS
      INTEGER ECON
      REAL MPGU,MPGCB
      REAL ROEPS,ROSPP,ROPPS,ROEPSL,ROPLL,ROES,RHOSS,BDPKWH,CP,CL,YLI
      REAL KP,MPGH,K,DP,RI
      REAL DR,IR,TX,KPR,SWCNT,SWMG,SWCVT,SPCCVT
      INTEGER TYPE,SYS,TYPESE,SYSSE
```

```
5730C  S U B R O U T I N E          C O N V P
5740C***********************************************
5750C
5760C     CONVENTIONAL VEHICLE PRINT
5770       REAL KPC,PTMG,CDTO
5780       REAL IF,LRC,ICF,MCMO,MIFHV,NLDF,NSCV
5790       INTEGER IPTO,NSY,NBC,IPSU,NYL,NSYS,NBCS
5800       INTEGER ECON
5810       REAL MPGU,MPGCB
5820       REAL ROEPS,ROSPP,ROPPS,ROEPSL,ROPLL,ROES,RHOSS,BDPKWH,CP,CL,YLI
5830       REAL KP,MPGH,K,DP,RI
5840       REAL DR,IR,TX,KPR,SWCNT,SWMG,SWCVT,SPCCVT
5850       INTEGER TYPE,SYS,TYPESE,SYSSE
5860       DIMENSION GAMMA(6),FDE(6)
5870       DIMENSION ROEPS(5),ROSPP(5),ROPPS(5),ROEPSL(5),ROPLL(5),ROES(6)
5880       DIMENSION RHOSS(5),BDPKWH(6),CP(6),CL(5),YLI(6),EE(5),BSV(5)
5890       DIMENSION RI(6),FHE(6),FSS(6)
5900       DIMENSION MIFHV(6)
5910       COMMON /INITV/SYS,TYPE,KP,ETAD,WI,K,CW,PW,KPC,
5920      & SPCHE,TMP,SEXU,SEXH,EFPHI,EFPUI,FUI,BSFCU,BSFCH,
5930      & SPCMG,SPCGEN,SPCCNT,SPCTRN,DPKWHE,ETAM,HBFI,SWENG,NYL,
5940      & SWTRN,ANC,GPO,MPGU,MPGCB,CHEF,MPGH,IPTO,NSY,NBC,IPSU,
5950      & ROEPS,ROSPP,ROPPS,ROEPSL,ROPLL,ROES,RHOSS,BDPKWH,CP,CL,YLI,EE,
5960      & FHE,FDE,FSS,YMI,DP,NSYS,NBCS,TYPESE,SYSSE,RI,GF,GK,CM,
5970      & DR,IR,NS,NF,TX,KPR,SWCVT,SPCCVT,SWCNT,SWMG,SWGEN
5980      & ,FMEDI,ETAPP,ECON,IF,NSCV,VICF,NLDF,ST,DMUP,CCV
5990      & ,MIFHV,LRC,ICF,MCMO,FINC,UBCC,BSV
6000C



6010       COMMON /INITN/SEX,WVC,ETAEU,ETAEH,WDTO,GAMMA
6020       COMMON /INITS/CDTO,DTMG,SEXHCV,OPCTG,TACCV,AGCCV,EFPU,EFPH,FU,YM
6030C
6040C END OF REAL/INTEGER/DIMENSION/COMMON STUFF
6050C
6060       WRITE(16,100)
6070   100 FORMAT(2X,//)
6080       WRITE(16,110)
6090   110 FORMAT(2X,'CONVENTIONAL VEHICLE',T26,'WVO(LBS)',T35,'WDTO(LBS)',
6100      & T46,'CDTO(%)',T55,'GPO($/GAL)',T66,'DTMG($/TON-MI)',T80
6110      & ,'OPCT(CTS/MI)',T94,'TACCV($/YR)',T110,'AGCCV')
6120       WRITE(16,120)WVO,WDTO,CDTO,GPO,DTMG,OPCTG,TACCV,AGCCV
6130   120 FORMAT(2X,T25,F9.0,T35,F9.0,T45,F9.2,T55,F9.2,T67,F9.2,T80
6140      & ,F9.2,T94,F9.2,T105,F9.2)
6150       RETURN
6160       END
6170       SUBROUTINE PUREP
6180C
6190C***********************************************
6200C  S U B R O U T I N E          P U R E P
6210C***********************************************
6220C
6230       INTEGER SYS
6240       COMMON /INITV/SYS
6250       WRITE(16,100)
6260   100 FORMAT(//,2X,'PURE STORAGE  :PRIMARY ONLY',5X,'PHI',7X,'RR',8X,
6270      & 'WV1R',5X,'WV',8X,'WDT1R',5X,'WDT',7X,'WB',4X,'YL')
```

```
C     SYS=1
C     ROEPS=WH/LD   (PRIMARY)
C     ROPPS=W/LD    (PRIMARY)
      CALL PRIMRY(SYS)
      RETURN
      END
      SUBROUTINE PARAHP
C
C*************************************************
C  S U B R O U T I N E        P A R A H P
C*************************************************
C
      INTEGER SYS
      COMMON /INITV/SYS
      WRITE(16,100)
  100 FORMAT(2X,'PARALLEL HYBRID:PRIMARY ONLY',5X,'PHI',7X,'RR',8X,
     & 'WV1R',5X,'WV',8X,'WDT1R',5X,'WDT',7X,'WB',4X,'YL')
      CALL PRIMRY(SYS)
      RETURN
      END
      SUBROUTINE PRIMRY(SYS)
C
C*************************************************
C  S U B R O U T I N E        P R I M R Y
C*************************************************
C
      REAL KPC,DTMG,CDTO
      REAL IF,LRC,ICF,MCMO,MIFHV,NLDF,NSCV
      INTEGER IPTO,NSY,NBC,IPSU,NYL,NSYS,NBCS
      INTEGER ECON
      REAL MPGU,MPGCB
      REAL ROEPS,ROSPP,ROPPS,ROEPSL,ROPLL,ROES,RHOSS,BDPKWH,CP,CL,YLI
      REAL KP,MPGH,K,DP,RI
      REAL DR,IR,TX,KPR,SWCNT,SWMG,SWCVT,SPCCVT
      REAL IRE
      INTEGER TYPE,SYS,TYPESE,SYSSE
      DIMENSION GAMMA(6),FDE(6),ALPHA(6)
      DIMENSION ROEPS(5),ROSPP(5),ROPPS(5),ROEPSL(5),ROPLL(5),ROES(6)
      DIMENSION RHOSS(5),BDPKWH(6),CP(6),CL(5),YLI(6),EE(5),BSV(5)
      DIMENSION RI(6),FHE(6),FSS(6)
      DIMENSION MIFHV(6)
      COMMON /INITV/ISYS,TYPE,KP,ETAD,WI,K,CW,PW,KPC,
     & SPCHE,TMP,SEXU,SEXH,EFPHI,EFPUI,FUI,BSFCU,BSFCH,
     & SPCMG,SPCGEN,SPCCNT,SPCTRN,DPKWHE,ETAM,HBFI,SWENG,NYL,
     & SWTRN,AMC,GPC,MPGU,MPGCR,CHFF,MPGH,IPTO,NSY,NBC,IPSU,
     & ROEPS,ROSPP,ROPPS,ROEPSL,ROPLL,ROES,RHOSS,BDPKWH,CP,CL,YLI,EE,
     & FHE,FDE,FSS,YMI,DP,NSYS,NBCS,TYPESE,SYSSE,RI,GF,GK,CM,
     & DR,IR,NS,NF,TX,KPR,SWCVT,SPCCVT,SWCNT,SWMG,SWGEN
     & ,FMEDI,ETAPP,ECON,IF,NSCV,VICF,NLDF,ST,DMUP,CCV
     & ,MIFHV,LRC,ICF,MCMO,FINC,UBCC,BSV
C
      COMMON /INITM/SEX,WVO,ETAEU,ETAEH,WDTO,GAMMA,YMP,ALPHA
      COMMON /INITS/CDTO,DTMG,SEXHCV,OPCTG,TACCV,AGCCV,EFPU,EFPH,FU,YM
C
      REAL IC,NHEUPG,MPG
      INTEGER TYPE,SYS
      DIMENSION PHIPR(5),PHIPP(5),PHI(5),RR(5)
      DIMENSION WV1R(5),WV(5),WDT1R(5),WDT(5),WB(5)
      DIMENSION YL(5),PHISPP(5)
```

```
      DIMENSION CPNEGP(5),DPTNEQ(5),DPTNEL(5),BEUPG(5),NBEUPG(5)
      DIMENSION DBEUPG(5)
      DIMENSION PC(5),CDT(5),CDTI(5),DELCW(5),IC(5),DELCDT(5)
      DIMENSION ROEPS1(5),CEHV(5)
C
C END OF REAL/INTEGER/DIMENSION/COMMON DESCRIPTIONS
C
      IF(SYS.NE.1.OR.SYS.NE.2)HBF=HBFI
      IF(SYS.NE.1.OR.SYS.NE.2)FU=FUI
      IF(SYS.NE.1.OR.SYS.NE.2)YM=YMI
      IF(SYS.NE.1.OR.SYS.NE.2)EFPU=EFPUI
      IF(SYS.NE.1.OR.SYS.NE.2)EFPH=EFPHI
      IF(SYS.NE.1.OR.SYS.NE.2) FMED=FMEDI
      IF(SYS.EQ.1.OR.SYS.EQ.2)FU=1.
      IF(SYS.EQ.1.OR.SYS.EQ.2) FMED=1.
      IF(SYS.EQ.1.OR.SYS.EQ.2)YM=YMP
      IF(FU.EQ.1.)HBF=1.
      IF(SYS.EQ.1.OR.SYS.EQ.2)EFPU=0.
      IF(SYS.EQ.1.OR.SYS.EQ.2)EFPH=1.
      RIO=RI(1)
      RIS=RI(SYS)
C
C***********  F I R S T   'DO' LOOP
C
      DO 2000 TYPE=NBCS,NBC
      ROEPS1(TYPE)=ROEPS(TYPE)*( RIS/RIO )** EE(TYPE)
      PHIPR(TYPE)=SEX*RIS/2./ETAD/ROEPS1(TYPE)
      PPDE=1.-FHE(SYS)
      IF (FDE(SYS).GT.PPDE) PPDE=FDE(SYS)
      PHIPP(TYPE)=PPDE*1000.*KP/ROPPS(TYPE)
      PHISPP(TYPE)=1000.*( GK-  (GF*KP*FHE(SYS)))/ROSPP(TYPE)
      IF(PHIPR(TYPE).GT.PHIPP(TYPE))PHI(TYPE)=PHIPR(TYPE)
      IF(PHIPP(TYPE).GT.PHIPR(TYPE))PHI(TYPE)=PHIPP(TYPE)
      IF(PHISPP(TYPE).GT.PHI(TYPE))PHI(TYPE)=PHISPP(TYPE)
      RR(TYPE)=RIO*( 2.*ETAD*ROEPS(TYPE)*PHI(TYPE) / (SEX*RIO) )
     & **(1./(1.-EE(TYPE)))
      WV1R(TYPE)=1./(1.-((ALPHA(SYS)*KP))-((1.+K)*PHI(TYPE)))
      WV(TYPE)=WV1R(TYPE)*W1
      WDT1R(TYPE)=WV1R(TYPE)-1.
      WDT(TYPE)=WDT1R(TYPE)*W1
      WB(TYPE)=WV(TYPE)*PHI(TYPE)
C     CYCLE LIFE
      IF(NYL.EQ.0)GO TO 110
      IF(NYL.EQ.1)GO TO 100
100   YL(TYPE)=2.*DP*CL(TYPE)*ROEPS(TYPE)*PHI(TYPE)*ETAD/YM/FU/HBF/SEX
      GO TO 120
110   YL(TYPE)=YLI(TYPE)
120   IF(TYPE.EQ.1)WRITE(16,130)PHI(TYPE),RP(TYPE),WV1R(TYPE),WV(TYPE),
     & WDT1R(TYPE),WDT(TYPE),WB(TYPE),YL(TYPE)
130   FORMAT( 5X,'BATTERY TYPE LEAD ACID',F10.3,F10.0,F10.2,F10.0,
     & F10.2,2F10.0,F7.2)
      IF(TYPE.EQ.2)WRITE(16,140)PHI(TYPE),RR(TYPE),WV1R(TYPE),WV(TYPE),
     & WDT1R(TYPE),WDT(TYPE),WB(TYPE),YL(TYPE)
140   FORMAT(5X,'BATTERY TYPE:PB ACID A',F10.3,F10.0,F10.2,F10.0,
     & F10.2,2F10.0,F7.2)
```

```
      IF(TYPE.EQ.3)WRITE(16,150)PHI(TYPE),RR(TYPE),WV1R(TYPE),WV(TYPE),
     & WDT1R(TYPE),WDT(TYPE),WB(TYPE),YL(TYPE)
  150 FORMAT(9X,'BATTERY TYPE:NI-ZN',F10.3,F10.0,F10.2,F10.0,
     & F10.2,2F10.0,F7.2)
      IF(TYPE.EQ.4)WRITE(16,160)PHI(TYPE),RR(TYPE),WV1R(TYPE),WV(TYPE),
     & WDT1R(TYPE),WDT(TYPE),WB(TYPE),YL(TYPE)
  160 FORMAT(9X,'BATTERY TYPE=NI FE',F10.3,F10.0,F10.2,F10.0,F10.2,
     & 2F10.0,F7.2)
      IF(TYPE.EQ.5)WRITE(16,170)PHI(TYPE),RR(TYPE),WV1R(TYPE),WV(TYPE),
     & WDT1R(TYPE),WDT(TYPE),WB(TYPE),YL(TYPE)
  170 FORMAT(11X,'BATTERY TYPE=L S',F10.3,F10.0,F10.2,F10.0,F10.2,
     & 2F10.0,F7.2)
 2000 CONTINUE
C
C HEADER FOR DRIVELINE COST
C
      WRITE(16,180)
  180 FORMAT(2X,///)
      WRITE(16,190)
  190 FORMAT(2X,'DRIVELINE COSTS')
      WRITE(16,210)
  210 FORMAT(2X,'PRIMARY',T17,'PRIMARY',T30,'SECONDARY'
     & ,T45,'SECONDARY',T65,'TOTAL',T84,'INCREMENTAL',T100,'EHV'
     & ,/,4X,'TYPE',T19,'COST',T33,'TYPE',T48,'COST'
     & ,T60,'DRIVELINE COST',T82,'DRIVELINE COST',T100,'COST')
C
C********** S E C O N D   'DO' LOOP
C
      DO 2001 TYPE=NBCS,NRC
C   PRIMARY CUST PC(TYPE)
      PC(TYPE)=WV(TYPE)*PHI(TYPE)*CP(TYPE)*(1.+DMUP)
      CBCH=(WV(TYPE)*PHI(TYPE)*ROEPS1(TYPE)*UBCC*(1.+DMUP)) / 1000.
C   TOTAL DRIVETRAIN COST CDT
      CDT(TYPE)=PC(TYPE)+(WV(TYPE)*KP*GAMMA(SYS))*(1.+DMUP) + CBCH
C   CONVENTIONAL COST CDTO (HEAT ENGINE+TRANS)
      CDTO=WVO*KPC*(SPCHE+SPCTRN)*(1.+DMUP)
C   THE INCREMENTAL DRIVELINE COST CDT1
      CDTI(TYPE)=CDT(TYPE)-CDTO
C   COST OF ADDITIONAL WEIGHT OF CAR AT AWC$/LB (DELCW)
      DELCW(TYPE)=K*PHI(TYPE)*WV(TYPE)*AWC*(1.+DMUP)
      IC(TYPE)=CDTI(TYPE)-PC(TYPE)*(1.-BSV(TYPE))
      DELCDT(TYPE)=IC(TYPE)+DELCW(TYPE)
C   CPMEQP= CENT PER MILE OF EQUIPMENT
C   DR=DISCOUNT RATE, IR=INTEREST RATE
C   NS=PAYBACK PERIOD STRUCTURE (YRS)
C   NF=FINANCIAL PERIOD (YRS)
```

```
7880C     YL=BATTERY LIFE (YRS)
7890C
7900C EVALUATE CHEVS & NLDF (IF NLDF NOT EQUAL TO 1 BY USER)
7910C
7920       IF(NLDF.NE.1.) NLDF= (2.*NS-NF+1.)/(NS+1.)
7930       CHEVS=DFEL(DT(TYPE) * (CCV-CDTO)*(1.+VICF)
7940       CEHV(TYPE)=CHEVS + PC(TYPE)*(1.-BSV(TYPE))
7950       IRE=(1.-TX)*IR
7960       FF=NF*IRE/(1.-(1.+IRE)**(-NF))
7970       FFO=FF * (DR-IF)/(1.+IF)
7980       FF1=(1.+DR)/(1.+IF)
7990       FRCS=FFO / (1.-FF1**(-NS))
8000       FRCB=FFO / (1.-FF1**(-YL(TYPE)))
8010       CPMEOP(TYPE)=0100./YM*((FRCS*NLDF*CHEVS) +
8020      & (FRCB*(1.-BSV(TYPE))*WV(TYPE)*PHI(TYPE)*CP(TYPE)))
8030C     DPTMEQ(TYPE)=DOLLARS PER TON MILES EQUIPMENT
8040       DPTMEQ(TYPE)=CPMEOP(TYPE)/100./(W1/2000.)
8050C     DPTMEL=DOLLARS PER TON MILES ELECTRIC
8060C     DPKWHE=DOLLARS PER KILOWATT HOUR ELECTRIC
8070       DPTMEL(TYPE)=(WV(TYPE)/W1)*DPKWHE*((FU*FMED*SEXU*(1.-EFPU)/ETAD)+
8080      & ((1.-FU)*SEXH*(1.-EFPH)/ETAD))/CHEF
8090C
8100       FRCV= FFO / (1. - FF1**(-NSCV))
8110       DFCV=(2.*NSCV-NF+1.) / (NSCV+1.)
8120       DTMCV=(FRCV*DFCV*CCV*(1.+ST)) / (YM*W1/2000.)
8130C
8140C LRC  = LICENSE AND REGISTER FEE
8150C FINC = FIXED INSURANCE COST
8160C ICF  = INSURANCE COST FACTOR
8170C MCMO = MAINTINENCE COST PER MILE FOR CV
8180C MIFHV = MAINTINANCE IMPROVEMENT FACTOR FOR HYBRID VEHICLE
8190C
8200       ADCHV=LRC+FINC+ICF*(CHEVS+PC(TYPE))
8210       ADCCV=LRC+FINC+ICF*CCV
8220       MCHV=.01*MCMO*YM*(1.-MIFHV(SYS))
8230       MCCV=.01*MCMO*YM
8240C
8250       NBEUPG(TYPE)=DPTMEQ(TYPE)+DPTMEL(TYPE)
8260      & + DTMCV + (ADCHV-ADCCV+MCHV-MCCV) / (YM*W1/2000.)
8270       IF(FU.EQ.1.)MPG=MPGU
8280       IF(FU.NE.1.)MPG=MPGU*MPGH / (FU*MPGH + (1.-FU)*MPGU)
8290       SEXHCV=35.2/MPG/WVO*2000.
8300       DTMG=GPO/MPG/W1*2000.
8310       DBEUPG(TYPE)=.03 * ( SEXHCV*WVO/W1 -
8320      & WV1R(TYPE)*( FU*FMED*SEXU*EFPU / (ETAEU*ETAM) +
8330      &              (1.-FU)*SEXH*EFPH / (ETAEH*ETAM) +
8340      &    FU*(1.-FMED)*SEXU / (ETAM*ETAEU) ) )
8350C     BEUPG= BREAKEVEN PRICE OF GASOLINE
8360       BEUPG(TYPE)=NBEUPG(TYPE)/DBEUPG(TYPE)
8370       IF(TYPE.EQ.1)WRITE(16,220)PC(TYPE),CDT(TYPE),CDTI(TYPE)
8380      & ,CEHV(TYPE)
8390  220 FORMAT(2X,'PB-ACID',T17,F10.0,T60,F10.0,T82,F10.0,T95,F10.0)
8400       IF(TYPE.EQ.2)WRITE(16,230)PC(TYPE),CDT(TYPE),CDTI(TYPE)
```

```
8410      & ,CFHV(TYPE)
8420   230 FORMAT(2X,'PB ACID A',T17,F' .0,T60,F10.0,T82,F10.0,T95,F10.0)
8430       IF(TYPE.EQ.3)WRITE(16,240)  (TYPE),CDT(TYPE),CDTI(TYPE)
8440      & ,CFHV(TYPE)
8450   240 FORMAT(2X,'NI ZN',T17,F'  .0,T60,F10.0,T82,F10.0,T95,F10.0)
8460       IF(TYPE.EQ.4)WRITE(16,250)PC(TYPE),CDT(TYPE),CDTI(TYPE)
8470      & ,CFHV(TYPE)
8480   250 FORMAT(2X,'NI FE',T17,F10.0,T60,F10.0,T82,F10.0,T95,F10.0)
8490       IF(TYPE.EQ.5)WRITE(16,260)PC(TYPE),CDT(TYPE),CDTI(TYPE)
8500      & ,CFHV(TYPE)
8510   260 FORMAT(2X,'LI-S',T1 ,F10.0,T60,F10.0,T82,F10.0,T95,F10.0)
8520  2001 CONTINUE
8530C
8540C HEADER FOR BREAK EVEN GAS PRICE
8550C
8560       WRITE(16,270)
8570   270 FORMAT(2X,///)
8580       WRITE(16,280)
8590   280 FORMAT(2X,'OPERATING COSTS AND BREAKEVEN GASOLINE PRICES')
8600       WRITE(16,290)
8610   290 FORMAT(T59,'GAS BKEV',T70,'OPER COST EQ&EL',T87,'HV GAS-PO'
8620      & ,T103,'DRIVE CHAR')
8630       WRITE(16,300)
8640   300 FORMAT(2X,'TYPE',T19,'DELCDT',T27,'BATTERY COST',T40,'DPTMEQ',
8650      & ,T50,'DPTMEL',T61,'$/GAL',T70,'CTS/MI',T80,'$/YR',T90,
8660      & '$/YR',T101,'LB/KW',T112,'$/KW')
8670C
8680C************** T H I R D    'DO' LOOP
8690C
8700       DO 2002 TYPE=NHCS,NBC
8710       GCPMHV=0.
8720       IF (FU.EQ.1.) GOTO 145
8730       GCPMHV=GPO*(WV(TYPE)/2000.)/(36.63*ETAM) *
8740      & ( SEXU*FU*FMED*EFPU / ETAEU + SEXU*FU*(1.-FMED) / ETAEU +
8750      &   SEXH*(1.-FU)*EFPH / ETAEH )
8760   145 CONTINUE
8770C
8780C  CALCULATE OPERATING COST PER MILE (CENTS/MI)
8790C
8800       OPCTEHV=100.*( (W1/2000.)*(DPTMEQ(TYPE)+DPTMEL(TYPE)) +
8810      &                (ADCHV+MCHV)/YM + GCPMHV )
8820       OPCTG=100.*( (W1/2000.)*(DTMCV+DTMG) + (ADCCV+MCCV)/YM )
8830C
8840C CALCULATE TOTAL ANNUAL OPERATING COST ($/YR)  TACEHV
8850C
8860       AOPCG=GCPMHV*YM
8870       TACEHV=OPCTEHV*(YM/100.)
8880       AOPCEQE=TACEHV-AOPCG-ADCHV-MCHV
8890       TACCV=OPCTG*(YM/100.)
8900       AGCCV=DTMG*(W1/2000.)*YM
8910       IF(TYPE.EQ.1)WRITE(16,310)DELCDT(TYPE),PC(TYPE),DPTMEQ(TYPE),
8920      & DPTMEL(TYPE),BEUPG(TYPE),OPCTEHV,AOPCEQE,AOPCG,ALPHA(SYS),
8930      & GAMMA(SYS)
8940   310 FORMAT(2X,'PB-ACID',T15,F10.0,T26,F10.0,T37,F9.3,T47,F9.3
8950      & ,T57,F9.2,T67,F9.2,T77,F9.2,T87,F9.2,T97,F9.2,T107,F9.2)
8960       IF(TYPE.EQ.2)WRITE(16,320)DELCDT(TYPE),PC(TYPE),DPTMEQ(TYPE),
8970      & DPTMEL(TYPE),BEUPG(TYPE),OPCTEHV,AOPCEQE,AOPCG,ALPHA(SYS),
8980      & GAMMA(SYS)
8990   320 FORMAT(2X,'PB ACID A',T15,F10.0,T26,F10.0,T37,F9.3,T47,F9.3
9000      & ,T57,F9.2,T67,F9.2,T77,F9.2,T87,F9.2,T97,F9.2,T107,F9.2)
```

```
      IF(TYPE.EQ.3)WRITE(16,330)DELCDT(TYPE),PC(TYPE),DPTMEQ(TYPE),
     & DPTMEL(TYPE),BEUPG(TYPE),OPCTEHV,AOPCEQE,AOPCG,ALPHA(SYS),
     & GAMMA(SYS)
  330 FORMAT(2X,'NI ZN',T15,F10.0,T26,F10.0,T37,F9.3,T47,F9.3
     & ,T57,F9.2,T67,F9.2,T77,F9.2,T87,F9.2,T97,F9.2,T107,F9.2)
      IF(TYPE.EQ.4)WRITE(16,340)DELCDT(TYPE),PC(TYPE),DPTMEQ(TYPE),
     & DPTMEL(TYPE),BEUPG(TYPE),OPCTEHV,AOPCEQE,AOPCG,ALPHA(SYS),
     & GAMMA(SYS)
  340 FORMAT(2X,'NI FE',T15,F10.0,T26,F10.0,T37,F9.3,T47,F9.3
     & ,T57,F9.2,T67,F9.2,T77,F9.2,T87,F9.2,T97,F9.2,T107,F9.2)
      IF(TYPE.EQ.5)WRITE(16,350)DELCDT(TYPE),PC(TYPE),DPTMEQ(TYPE),
     & DPTMEL(TYPE),BEUPG(TYPE),OPCTEHV,AOPCEQE,AOPCG,ALPHA(SYS),
     & GAMMA(SYS)
  350 FORMAT(2X,'LI-S',T15,F10.0,T26,F10.0,T37,F9.3,T47,F9.3
     & ,T57,F9.2,T67,F9.2,T77,F9.2,T87,F9.2,T97,F9.2,T107,F9.2)
 2002 CONTINUE
C
C OUTPUT ENERGY CALCULATIONS IF ECON=1
C
      NTYPE=NBC-NBCS+1
      IF (ECON.EQ.1) CALL ENCAL(WV(NBCS),NTYPE)
C
C OUTPUT CONVENTIONAL VEHICLE INFO
C
      CALL CONVP
      RETURN
      END
      SUBROUTINE PSECDY(SYS)
C
C**********************************************
C  S U B R O U T I N E        P S E C D Y
C**********************************************
C
      REAL KPC,DTMG,CDTO
      REAL IF,LRC,ICF,MCMO,MIFHV,NLDF,NSCV
      INTEGER IPTO,NSY,NBC,IPSU,NYL,NSYS,NBCS
      INTEGER ECON
      REAL MPGU,MPGCB
      REAL ROEPS,ROSPP,ROPPS,ROEPSL,ROPLL,ROES,RHOSS,BDPKWH,CP,CL,YLI
      REAL KP,MPGH,K,DP,RI
      REAL DR,IR,TX,KPR,SWCNT,SWMG,SWCVT,SPCCVT
      REAL IRT
      INTEGER TYPE,SYS,TYPESE,SYSSE
      DIMENSION GAMMA(6),FDE(6),ALPHA(6)
      DIMENSION ROEPS(5),ROSPP(5),ROPPS(5),ROEPSL(5),ROPLL(5),ROES(6)
      DIMENSION RHOSS(5),BDPKWH(6),CP(6),CL(5),YLI(6),EE(5),BSV(5)
      DIMENSION RI(6),FHE(6),FSS(6)
      DIMENSION MIFHV(6)
      COMMON /INITV/ISYS,TYPE,KP,ETAD,WI,K,CW,PW,KPC,
     & SPCHE,TMP,SEXU,SEXH,EFPHI,EFPUI,FUI,BSFCU,BSFCH,
     & SPCMG,SPCGEN,SPCCNT,SPCTRN,DPKWHE,ETAM,HBFI,SWENG,NYL,
     & SWTRN,AWC,GPO,MPGU,MPGCB,CHEF,MPGH,IPTO,NSY,NBC,IPSU,
     & ROEPS,ROSPP,ROPPS,ROEPSL,ROPLL,ROES,RHOSS,BDPKWH,CP,CL,YLI,EE,
```

```
9540      & FHE,FDE,FSS,YMI,DP,NSYS,NBCS,TYPESE,SYSSE,RI,GF,GK,CM,
9550      & DR,IR,NS,NF,TX,KPR,SWCVT,SPCCVT,SWCNT,SWMG,SWGEN
9560      & ,FMEDI,ETAPP,ECON,IF,NSCV,VICF,NLDF,ST,DMUP,CCV
9570      & ,MIFHV,LRC,ICF,MCMO,FINC,UBCC,USV
9580C
9590       COMMON /INITM/SEX,WVO,ETAEU,ETAEH,WDTO,GAMMA,YMP,ALPHA
9600       COMMON /INITS/CDTO,DTMG,SEXHCV,OPCTG,TACCV,AGCCV,EFPU,EFPH,FU,YM
9610C
9620       REAL KPSS,MPG
9630       DIMENSION CLS(4)
9640       REAL IC,NBEUPG
9650       INTEGER TYPE
9660       INTEGER SYS
9670       DIMENSION A(4),CA(4)
9680       DIMENSION AL(4),CAL(4)
9690       DIMENSION ALP(4),CALP(4)
9700       DIMENSION REPSL(4),PHIPR(4),PHIPP(4),RPLL(4),PHIP(4),YL(4)
9710       DIMENSION PHISS(4),PHIT(4),RHOS(2),RR(4),WV1R(4),WV(4)
9720       DIMENSION WDT1R(4),WDT(4),WP(4),WS(4)
9730       DIMENSION SC(4),PC(4),BC(4),CDT(4),CPPS(4),CSPS(4),CDTI(4),DEL
9740      & CW(4)
9750       DIMENSION IC(4),DELCDT(4),CPMEQP(4),DPTMEL(4),DPTMEQ(4),BEUPG(4)
9760       DIMENSION NBEUPG(4),DBEUPG(4)
9770       DIMENSION REPSL1(4),CEHV(4)
9780C
9790C END OF REAL/INTEGER/DIMENSION STUFF
9800C
9810       YM=YMI
9820       FU=FUI
9830       HBF=HBFI
9840       EFPU=EFPUI
9850       EFPH=EFPHI
9860       FMED=FMEDI
9870       CLS(1)=CL(1)
9880       CLS(2)=CL(1)
9890       CLS(3)=CL(5)
9900       CLS(4)=CL(5)
9910C     SYS=2  PURE ELECTRIC-P/S
9920C     SYS=3  SERIES-P/S
9930C     SYS=5  PARALLEL-P/S
9940       IF(SYS.EQ.2) GO TO 100
9950       IF(SYS.EQ.3) GO TO 120
9960       IF(SYS.EQ.5) GO TO 140
9970   100 WRITE(16,110)
```

```
 9980  110 FORMAT(2X,'PURE STORAGE :PRIMARY AND SECONDARY',4X,'PHIT',6X,
 9990     & 'RR',8X,'WV1R',5X,'WV',8X,'WDT',7X,'WP',8X,'WS',8X,'ALPHA')
10000C     SYS=2
10010       IF(SYS.EQ.2)FU=1.
10020       IF(FU.EQ.1.)HBF=1.
10030       IF(SYS.EQ.2)EFPU=0.
10040       IF(SYS.EQ.2)EFPH=1.
10050       IF(SYS.EQ.2)YM=YMP
10060       IF(SYS.EQ.2) FMED=1.
10070       GO TO 160
10080  120 WRITE(16,130)
10090  130 FORMAT(2X,'SERIES HYBRID  :PRIMARY AND SECONDARY',4X,'PHIT',6X,
10100     & 'RR',8X,'WV1R',5X,'WV',8X,'WDT',7X,'WP',8X,'WS',8X,'ALPHA')
10110C     SYS=3
10120       GO TO 160
10130  140 WRITE(16,150)
10140  150 FORMAT(2X,'PARALLEL HYBRID:PRIMARY AND SECONDARY',4X,'PHIT',6X,
10150     & 'RR',8X,'WV1R',5X,'WV',8X,'WDT',7X,'WP',8X,'WS',8X,'ALPHA')
10160C     SYS=5
10170       GO TO 160
10180  160 CONTINUE
10190       REPSL(1)=ROEPSL(1)
10200       REPSL(2)=ROEPSL(1)
10210       REPSL(3)=ROEPSL(5)
10220       REPSL(4)=ROEPSL(5)
10230       RPLL(1)=ROPLL(1)
10240       RPLL(2)=ROPLL(1)
10250       RPLL(3)=ROPLL(5)
10260       RPLL(4)=ROPLL(5)
10270       RIO=RI(1)
10280       RIS=RI(SYS)
10290C
10300C************ F I R S T   'DO' LOOP
10310C
10320       DO 2000 TYPE=1,4
10330       REPSL1(TYPE)=REPSL(TYPE)*( RIS/RIO )**EE(TYPE)
10340       PHIPR(TYPE)=SLX*RIS/2./ETAD/REPSL1(TYPE)
10350       PHIPP(TYPE)=1000.*(GK   - (GF*KPR*FHE(SYS)))/RPLL(TYPE)
10360       IF(PHIPP(TYPE).GT.PHIPR(TYPE))PHIP(TYPE)=PHIPP(TYPE)
10370       IF(PHIPR(TYPE).GT.PHIPP(TYPE))PHIP(TYPE)=PHIPR(TYPE)
10380       RHOS(1)=RHOSS(1)
10390       RHOS(2)=ROES(6)
10400       KPSS=KF-(PHIP(TYPE)*RPLL(TYPE)/1000.)-(FHE(SYS)*KPR)
10410       IF (KPSS.LT.0.) KPSS=0.
10420       PHISS(1)=KPSS*1000./RHOS(1)
10430       PHISS(2)=KPSS*TMP/3.6/RHOS(2)
10440       PHISS(3)=PHISS(1)
10450       PHISS(4)=PHISS(2)
10460       IF (SYS.EQ.2) GO TO 170
10470       IF (SYS.EQ.3) GO TO 180
10480       IF (SYS.EQ.5) GO TO 190
10490  170 A(TYPE)=ALPHA(SYS)
10500       IF (TYPE.EQ.2.OR.TYPE.EQ.4) A(TYPE)=A(TYPE)+(KPSS/KP)*SWCVT
10510       CA(TYPE)=GAMMA(SYS)
10520       IF (TYPE.EQ.2.OR.TYPE.EQ.4) CA(TYPE)=CA(TYPE)+(KPSS/KP)*SPCCVT
10530       ALPHSS=IF(TYPE)
```

```
      ALPHSS=ALF(TYPE)
      GO TO 200
  180 AL(TYPE)=ALPHA(SYS)
      IF (TYPE.EQ.2.OR.TYPE.EQ.4) AL(TYPE)=AL(TYPE)+(KPSS/KP)*SWCVT
      CAL(TYPE)=GAMMA(SYS)
      IF (TYPE.EQ.2.OR.TYPE.EQ.4) CAL(TYPE)=CAL(TYPE)+(KPSS/KP)*SPCCVT
      ALPHSS=AL(TYPE)
      GO TO 200
  190 ALP(TYPE)=ALPHA(SYS)
      IF (TYPE.EQ.2.OR.TYPE.EQ.4) ALP(TYPE)=ALP(TYPE)+(KPSS/KP)*SWCVT
      CALP(TYPE)=GAMMA(SYS)
      IF (TYPE.EQ.2.OR.TYPE.EQ.4) CALP(TYPE)=CALP(TYPE)+(KPSS/KP)*SPCCVT
      ALPHSS=ALP(TYPE)
      GO TO 200
  200 CONTINUE
      PHIT(TYPE)=PHIP(TYPE)+PHISS(TYPE)
      RR(TYPE)=RIO*( 2.*ETAD*REPSL(TYPE)*PHIP(TYPE)/ (SEX*RIO) )
     & **(1./(1.-EE(TYPE)))
      WV1R(TYPE)=1./(1.-(ALPHSS*KP)-((1.+K)*PHIT(TYPE)))
      WV(TYPE)=W1*WV1R(TYPE)
      WDT1R(TYPE)=WV1R(TYPE)-1.
      WDT(TYPE)=WDT1R(TYPE)*W1
      WP(TYPE)=WV(TYPE)*PHIP(TYPE)
      WS(TYPE)=WV(TYPE)*PHISS(TYPE)
      IF(NYL.EQ.0)GO TO 220
      IF(NYL.EQ.1)GO TO 210
  210 YL(TYPE)=2.*DP*CLS(TYPE)*REPSL(TYPE)*PHIP(TYPE)*ETAD/YM/FU/HBF/SEX
      GO TO 230
```

```
  220 YL(1)=YLI(1)
      YL(2)=YLI(1)
      YL(3)=YLI(5)
      YL(4)=YLI(5)
  230 IF(TYPE.EQ.1)WRITE(16,240)PHIT(TYPE),RR(TYPE),WV1R(TYPE),WV(TYPE),
     & WDT(TYPE),WP(TYPE),WS(TYPE),ALPHSS
  240 FORMAT(17X,'LEAD ACID/LEAD ACID',F10.3,F10.0,F10.2,4F10.0,F10.2)
      IF(TYPE.EQ.2)WRITE(16,250)PHIT(TYPE),RR(TYPE),WV1R(TYPE),WV(TYPE),
     & WDT(TYPE),WP(TYPE),WS(TYPE),ALPHSS
  250 FORMAT(17X,'LEAD ACID/FLYWHEEL ',F10.3,F10.0,F10.2,4F10.0,F10.2)
      IF(TYPE.EQ.3)WRITE(16,260)PHIT(TYPE),RR(TYPE),WV1R(TYPE),WV(TYPE),
     & WDT(TYPE),WP(TYPE),WS(TYPE),ALPHSS
  260 FORMAT(19X,'HI TEMP/LEAD ACID',F10.3,F10.0,F10.2,4F10.0,F10.2)
      IF(TYPE.EQ.4)WRITE(16,270)PHIT(TYPE),RR(TYPE),WV1R(TYPE),WV(TYPE),
     & WDT(TYPE),WP(TYPE),WS(TYPE),ALPHSS
  270 FORMAT(20X,'HI TEMP/FLYWHEEL',F10.3,F10.0,F10.2,4F10.0,F10.2)
 2000 CONTINUE
C
C HEADER FOR DRIVELINE COST
C
      WRITE(16,280)
  280 FORMAT(2X,///)
C    ECONOMIC MODEL
      CPPS(1)=CP(1)
C    LEAD ACID
      CPPS(2)=CP(1)
```

```
11070C     LEAD ACID
11080       CPPS(3)=CP(5)
11090C     LI S
11100       CPPS(4)=CP(5)
11110C     LI S
11120       CSPS(1)=CP(1)
11130C     LEAD ACID
11140       CSPS(2)=CP(6)
11150C     FLYWHEEL
11160       CSPS(3)=CP(1)
11170       CSPS(4)=CP(6)
11180       WRITE(16,290)
11190   290 FORMAT(2X,'DRIVELINE COSTS')
11200       WRITE(16,310)
11210   310 FORMAT(2X,'PRIMARY',T17,'PRIMARY',T30,'SECONDARY'
11220      & ,T45,'SECONDARY',T65,'TOTAL',T84,'INCREMENTAL',T100,'EHV'
11230      & ,/,4X,'TYPE',T19,'COST',T33,'TYPE',T48,'COST'
11240      & ,T60,'DRIVELINE COST',T82,'DRIVELINE COST',T100,'COST')
11250C
11260C********** S E C O N D   'DO' LOOP
11270C
11280       DO 2001 TYPE=1,4
11290       IF(SYS.EQ.2)GAMMSS=CA(TYPE)
11300       IF(SYS.EQ.3)GAMMSS=CAL(TYPE)
11310       IF(SYS.EQ.5)GAMMSS=CALP(TYPE)
11320C     PRIMARY COST PC
11330       PC(TYPE)=WV(TYPE)*PHIP(TYPE)*CPPS(TYPE)*(1.+DMUP)
11340       CBCH=(WV(TYPE)*PHIP(TYPE)*REPSL1(TYPE)*UBCC*(1.+DMUP)) / 1000.
11350C     SECONDARY COST SC
11360       SC(TYPE)=WV(TYPE)*PHISS(TYPE)*CSPS(TYPE)*(1.+DMUP)
11370C     BATTERY COST BC
11380       BC(TYPE)=PC(TYPE)
11390C     TOTAL DRIVETRAIN COST CDT
11400       CDT(TYPE)=PC(TYPE)+SC(TYPE)+(WV(TYPE)*KP*GAMMSS)*(1.+DMUP)
11410      & + CBCH
11420C     CONVENTIONAL COST CDTO(HEAT ENGINE+TRANS)
11430       CDTO=WVU*KPC*(SPCHE+SPCTRN)*(1.+DMUP)
11440C     THE INCREMENTAL DRIVELINE COST CDTI
11450       CDTI(TYPE)=CDT(TYPE)-CDTO
```

```
      CDTI(TYPE)=CDT(TYPE)-CDTO
C     COST OF ADDITIONAL WEIGHT OF CAR AT AWC$/LB
      DELCW(TYPE)=K*PHIT(TYPE)*WV(TYPE)*AWC*(1.+DMUP)
      IC(TYPE)=CDTI(TYPE)-BC(TYPE)*(1.-BSV(TYPE))
      DELCDT(TYPE)=IC(TYPE)+DELCW(TYPE)
C
C EVALUATE CHEVS & NLDF (IF NLDF NOT EQUAL TO 1 BY USER)
C
      IF(NLDF.NE.1.) NLDF= (2.*NS-NF+1.)/(NS+1.)
      CHEVS=DELCDT(TYPE) + (CCV-CDTO)*(1.+VICF)
      CEHV(TYPE)=CHEVS + BC(TYPE)*(1.-BSV(TYPE))
      IRE=(1.-TX)*IR
      FF=NF*IRE/(1.-(1.+IRE)**(-NF))
      FFO=FF * (DR-IF)/(1.+IF)
      FF1= (1.+DR)/(1.+IF)
      FRCS=FFO / (1.-FF1**(-NS))
      FRCB=FFO / (1.-FF1**(-YL(TYPE)))
      CPMEOP(TYPE)=0100./YM*((FRCS*NLDF*CHEVS) +
     & (FRCB*(1.-BSV(TYPE))*BC(TYPE)))*(1.+ST)
      DPTMEO(TYPE)=CPMEOP(TYPE)/100./(W1/2000.)
C     DPKWHE=DOLLARS PER KILOWATT HOUR ELECTRIC
C     DPTMEL=DOLLARS PER TON MILE ELECTRIC
      DPTMEL(TYPE)=(WV(TYPE)/W1)*DPKWHE*((FU*FMED*SEXU*(1.-EFPU)/ETAD)+
     & ((1.-FU)*SEXH*(1.-EFPH)/ETAD))/CHEF
C
      FRCV= FFO / (1.-FF1**(-NSCV))
      DFCV=(2.*NSCV-NF+1.) / (NSCV+1.)
      DTMCV=(FRCV*DFCV*CCV*(1.+ST)) / (YM*W1/2000.)
C
C LRC  = LICENSE AND REGISTER FEE
C FINC = FIXED INSURANCE COST
C ICF  = INSURANCE COST FACTOR
C MCMO = MAINTINENCE COST PER MILE FOR CV
C MIFHV = MAINTINANCE IMPROVEMENT FACTOR FOR HYBRID VEHICLE
C
      ADCHV=LRC+FINC+ICF*(CHEVS+BC(TYPE))
      ADCCV=LRC+FINC+ICF*CCV
      MCHV=.01*MCMO*YM*(1.-MIFHV(SYS))
      MCCV=.01*MCMO*YM
C
      NBEUPG(TYPE)=DPTMEO(TYPE)+DPTMEL(TYPE)
     & - DTMCV + (ADCHV-ADCCV+MCHV-MCCV) / (YM*W1/2000.)
      IF(FU.EQ.1.)MPG=MPGU
      IF(FU.NE.1.)MPG=MPGU*MPGH / (FU*MPGH + (1.-FU)*MPGU)
      SEXHCV=35.2/MPG/W1*2000.
      DTMG=GPU/MPG/W1*2000.
      DBEUPG(TYPE)=.03 * ( SEXHCV*WVO/W1 -
     & WV1R(TYPE)*( FU*FMED*SEXU*EFPU / (ETAEU*ETAM) +
     &         (1.-FU)*SEXH*EFPH / (ETAEH*ETAM) +
     &    FU*(1.-FMED)*SEXU / (ETAM*ETAEU) ) )
      BEUPG(TYPE)=NBEUPG(TYPE)/DBEUPG(TYPE)
C     BEUPG=BREAK EVEN UNIT PRICE OF GASOLINE
      IF(TYPE.EQ.1)WRITE(16,320)PC(TYPE),SC(TYPE),CDT(TYPE),CDTI(TYPE)
     & ,CEHV(TYPE)
  320 FORMAT(2X,'PB-ACID',T17,F10.0,T30,'PB-ACID',T45,F10.0,T60,F10.0,
     & T82,F10.0,T95,F10.0)
```

```
12010       IF(TYPE.EQ.2)WRITE(16,330)PC(TYPE),SC(TYPE),CDT(TYPE),CDTI(TYPE)
12020      & ,CEHV(TYPE)
12030   330 FORMAT(2X,'PB-ACID',T17,F10.0,T30,'FLYWHEEL',T45,F10.0,T60,F10.0,
12040      & T82,F10.0,T95,F10.0)
12050       IF(TYPE.EQ.3)WRITE(16,340)PC(TYPE),SC(TYPE),CDT(TYPE),CDTI(TYPE)
12060      & ,CEHV(TYPE)
12070   340 FORMAT(2X,'LI-S',T17,F10.0,T30,'PB-ACID',T45,F10.0,T60,F10.0,
12080      & T82,F10.0,T95,F10.0)
12090       IF(TYPE.EQ.4)WRITE(16,350)PC(TYPE),SC(TYPE),CDT(TYPE),CDTI(TYPE)
12100      & ,CEHV(TYPE)
12110   350 FORMAT(2X,'LI-S',T17,F10.0,T30,'FLYWHEEL',T45,F10.0,T60,
12120      & F10.0,T82,F10.0,T95,F10.0)
12130  2001 CONTINUE
12140C
12150C HEADER FOR BREAK EVEN GAS PRICES
12160C
12170       WRITE(16,360)
12180   360 FORMAT(2X,///)
12190       WRITE(16,370)
12200   370 FORMAT(2X,'OPERATION COSTS AND BREAKEVEN GASOLINE PRICES')
12210       WRITE(16,380)
12220   380 FORMAT(T59,'GAS BKEV',T70,'OPER COST EQ&EL',T87,'HV GAS-PO'
12230      & ,T103,'DRIVE CHAR')
12240       WRITE(16,390)
12250   390 FORMAT(2X,'TYPE',T19,'DELCDT',T27,'BATTERY COST',T40,'DPTMEQ',
12260      & ,T50,'DPTMEL',T61,'$/GAL',T70,'CTS/MI',T80,'$/YR',T90,
12270      & '$/YR',T101,'LB/KW',T112,'$/KW')
12280C
12290C*************** T H I R D   'DO' LOOP
12300C
12310       DO 2002 TYPE=1,4
12320       IF(SYS.EQ.2) GAMMSS=GA(TYPE)
12330       IF(SYS.EQ.3) GAMMSS=GAL(TYPE)
12340       IF(SYS.EQ.5) GAMMSS=GALP(TYPE)
12350       IF(SYS.EQ.2) ALPHSS=A(TYPE)
12360       IF(SYS.EQ.3) ALPHSS=AL(TYPE)
12370       IF(SYS.EQ.5) ALPHSS=ALP(TYPE)
12380       GCPMHV=0.
12390       IF (FU.EQ.1.) GOTO 145
12400       GCPMHV=GPO*(WV(TYPE)/2000.)/(36.63*ETAM) *
12410      & ( SEXU*FU*FMED*EFPU / ETAEU + SEXU*FU*(1.-FMED) / ETAEU +
12420      &   SEXH*(1.-FU)*EFPH / ETAEH )
12430   145 CONTINUE
12440C
12450C  CALCULATE OPERATING COST PER MILE (CENTS/MI)
12460C
12470       OPCTEHV=100.*( (W1/2000.)*(DPTMEQ(TYPE)+DPTMEL(TYPE)) +
12480      &                (ADCHV+MCHV)/YM + GCPMHV )
12490       OPCTG=100.*( (W1/2000.)*(DTMCV+DTMG) + (ADCCV+MCCV)/YM )
12500C
12510C CALCULATE TOTAL ANNUAL OPERATING COST ($/YR)  TACEHV
12520C
12530       AOPCG=GCPMHV*YM
12540       TACEHV=OPCTEHV*(YM/100.)
12550       AOPCEQE=TACEHV-AOPCG-ADCHV-MCHV
12560       TACCV=OPCTG*(YM/100.)
12570       AGCCV=DTMG*(W1/2000.)*YM
12580       IF(TYPE.EQ.1)WRITE(16,400)DELCDT(TYPE),BC(TYPE),DPTMEQ(TYPE),
```

```
12590      & DPTMEL(TYPE),BEUPG(TYPE),OPCTEHV,AOPCEQE,AOPCG,ALPHSS,GAMMSS
12600   400 FORMAT(2X,'PB-ACID',T15,F10.0,T26,F10.0,T37,F9.3,T47,F9.3
12610      & ,T57,F9.2,T67,F9.2,T77,F9.2,T87,F9.2,T97,F9.2,T107,F9.2)
12620       IF(TYPE.EQ.1)WRITE(16,410)
12630   410 FORMAT(2X,'PB-ACID')
12640       IF(TYPE.EQ.2)WRITE(16,420)DELCDT(TYPE),BC(TYPE),DPTMEQ(TYPE),
12650      & DPTMEL(TYPE),BEUPG(TYPE),OPCTEHV,AOPCEQE,AOPCG,ALPHSS,GAMMSS
12660   420 FORMAT(2X,'PB-ACID',T15,F10.0,T26,F10.0,T37,F9.3,T47,F9.3,T65
12670      & ,T57,F9.2,T67,F9.2,T77,F9.2,T87,F9.2,T97,F9.2,T107,F9.2)
12680       IF(TYPE.EQ.2)WRITE(16,430)
12690   430 FORMAT(2X,'FLYWHEEL')
12700       IF(TYPE.EQ.3)WRITE(16,440)DELCDT(TYPE),BC(TYPE),DPTMEQ(TYPE),
12710      & DPTMEL(TYPE),BEUPG(TYPE),OPCTEHV,AOPCEQE,AOPCG,ALPHSS,GAMMSS
12720   440 FORMAT(2X,'LI-S/PB-ACID',T15,F10.0,T26,F10.0,T37,F9.3,T47,F9.3
12730      & ,T57,F9.2,T67,F9.2,T77,F9.2,T87,F9.2,T97,F9.2,T107,F9.2)
12740       IF(TYPE.EQ.4)WRITE(16,450)DELCDT(TYPE),BC(TYPE),DPTMEQ(TYPE),
12750      & DPTMEL(TYPE),BEUPG(TYPE),OPCTEHV,AOPCEQE,AOPCG,ALPHSS,GAMMSS
12760   450 FORMAT(2X,'LI-S',T15,F10.0,T26,F10.0,T37,F9.3,T47,F9.3
12770      & ,T57,F9.2,T67,F9.2,T77,F9.2,T87,F9.2,T97,F9.2,T107,F9.2)
12780       IF(TYPE.EQ.4)WRITE(16,460)
12790   460 FORMAT(2X,'FLYWHEEL')
12800  2002 CONTINUE
12810C
12820C OUTPUT ENERGY CALCULATIONS IF ECON=1
12830C
12840       NTYPE=4
12850       IF (ECON.EQ.1) CALL ENCAL(WV(1),NTYPE)
12860C
12870C OUTPUT CONVENTIONAL VEHICLE STUFF
12880C
12890       CALL CONVP
12900       RETURN
12910       END
12920       SUBROUTINE ENCAL(WV,NTYPE)
12930C
12940C*******************************************
12950C  S U B R O U T I N E   E N C A L
12960C*******************************************
12970C
12980       DIMENSION WV(1)
12990       REAL KPC,DTMG,CDTO
13000       INTEGER IPTO,NSY,NBC,IPSU,NYL,NSYS,NBCS
13010       INTEGER ECON
13020       REAL MPGU,MPGCB
13030       REAL ROEPS,ROSPP,ROPPS,ROEPSL,ROPLL,ROES,RHOSS,BDPKWH,CP,CL,YLI
13040       REAL KP,MPGH,K,DP,RI
13050       REAL DR,IR,TX,KPR,SWCNT,SWMG,SWCVT,SPCCVT
13060       REAL IRE
```

```
13070       REAL IF,LRC,ICF,MCMO,MIFHV,NLDF,NSCV
13080       INTEGER TYPE,SYS,TYPESE,SYSSE
13090       DIMENSION GAMMA(6),FDE(6),ALPHA(6)
13100       DIMENSION ROEPS(5),ROSPP(5),ROPPS(5),ROEPSL(5),ROPLL(5),ROES(6)
13110       DIMENSION RHOSS(5),BDPKWH(6),CP(6),CL(5),YLI(6),EE(5),BSV(5)
13120       DIMENSION RI(6),FHE(6),FSS(6)
13130       DIMENSION MIFHV(6)
13140       COMMON /INITV/SYS,TYPE,KP,ETAD,W1,K,CW,PW,KPC,
13150      & SPCHE,IMP,SEXU,SEXH,EFPH1,EFPU1,FU1,BSFCU,BSFCH,
13160      & SPCMG,SPCGEN,SPCCNT,SPCTRN,DPKWHE,ETAM,HBFI,SWENG,NYL,
13170      & SWTRN,AWC,GPO,MPGU,MPGCB,CHEF,MPGH,IPTO,NSY,NBC,IPSU,
13180      & ROEPS,ROSPP,ROPPS,ROEPSL,ROPLL,ROES,RHOSS,BDPKWH,CP,CL,YLI,EE,
13190      & FHE,FDE,FSS,YMI,DP,NSYS,NBCS,TYPESE,SYSSE,RI,GF,GK,CM,
13200      & DR,IR,NS,NF,TX,KPR,SWCVT,SPCCVT,SWCNT,SWMG,SWGEN
```

```
13210      & ,FMEDI,ETAPP,ECON,IF,NSCV,VICF,NLDF,ST,DMUP,CCV
13220      & ,MIFHV,LRC,ICF,MCMO,FINC,UBCC,BSV
13230C
13240       COMMON /INITM/SEX,WVO,ETAEU,ETAEH,WDTO,GAMMA,YMP,ALPHA
13250       COMMON /INITS/CDTO,DTMG,SEXHCV,OPCTG,TACCV,AGCCV,EFPU,EFPH,FU,YM
13260C
13270       FMED=FMEDI
13280       IF(SYS.EQ.1.OR.SYS.EQ.2) FMED=1.
13290C
13300C END OF REAL/INTEGER/DIMENSION/COMMON STUFF
13310C
13320       WRITE(16,95)
13330    95 FORMAT(2X,///)
13340       WRITE(16,100)
13350   100 FORMAT(2X,'ENERGY RESULTS:',T25,'CEUM(KWH/MI)',T40
13360      & ,'CEHM(KWH/MI)',T55,'DLGLU(GAL/YR)',T70,'DLGLH(GAL/YR)')
13370       DO 1000 I=1,NTYPE
13380       EELU=(WV(I)/2000.)*( SEXU*FMED*(1.-EFPU)/(CHEF*ETAD*ETAPP) )
13390C
13400       DLGLU=(WV(I)/2000.)*( SEXU*FMED*EFPU/(ETAM*ETAEU) +
13410      &            SEXU*(1.-FMED)/(ETAM*ETAEU) ) - 36.63/MPGU
13420C
13430       DLGGU=DLGLU*FU*YM / 36.63
13440C
13450C
13460C  IF SYS=1 OR 2 THEN NO CALC FOR HIGHWAY
13470C
13480       EELH=0.
13490       DLGLH=0.
13500       DLGGH=0.
13510       IF(SYS.EQ.1.OR.SYS.EQ.2) GOTO 200
13520       EELH=(WV(I)/2000.)*( SEXH*(1.-EFPH)/(CHEF*ETAD*ETAPP) )
13530       DLGLH=(WV(I)/2000.)* (SEXH*EFPH/(ETAM*ETAEH)) - 36.63/MPGH
13540       DLGGH=DLGLH*(1.-FU)*YM / 36.63
13550   200 CONTINUE
13560       CEUM=EFPU*DLGLU
13570       CEHM=EELH*DLGLH
13580C
13590C
```

```
      WRITE(16,105) CEUM,CEHM,DLGGU,DLGGH
  105 FORMAT(1H5,F10.3,T40,F10.3,T55,F8.1,T70,F8.1)
C
 1000 CONTINUE
      RETURN
      END
C
C*****************************************
C        B L O C K   D A T A   S U B P R O G R A M
C*****************************************
C
C TO SET UP INITIAL DEFAULT VALUES
C    4/30/79
C
      BLOCK DATA
      REAL KPC,DTMG,CDTO
      REAL IF,LRC,ICF,MCMO,MIFHV,NLDF,NSCV
      INTEGER IPTO,NSY,NBC,IPSU,NYL,NSYS,NBCS
      INTEGER ECON
      REAL MPGU,MPGCB
      REAL ROEPS,ROSPP,ROPPS,ROEPSL,ROPLL,ROES,RHOSS,BDPKWH,CP,CL,YLI
      REAL KP,MPGH,K,YM,DP,RI
      REAL DR,IR,TX,KPR,SWCNT,SWMG,SWCVT,SPCCVT
      INTEGER TYPE,SYS,TYPESF,SYSSE
      DIMENSION GAMMA(6),FDE(6),ALPHA(6)
      DIMENSION ROEPS(5),ROSPP(5),ROPPS(5),ROEPSL(5),ROPLL(5),ROES(6)
      DIMENSION RHOSS(5),BDPKWH(6),CP(6),CL(5),YLI(6),EE(5),BSV(5)
      DIMENSION RI(6),FHE(6),FSS(6)
      DIMENSION MIFHV(6)
C
C****  COMMON BLOCKS ARE SET UP FOR USE IN THE SUBROUTINES
C****  SEHYPR, CONVP, PRIMRY, PSECDY, ENCAL
C****  COMMON  "INITV"  REFERS TO VARIABLES FROM INPUT DEVICE
C****  COMMON  "INITM"  REFERS TO VARIABLES INITIALIZED IN MAIN
C****  COMMON  "INITS"  REFERS TO VARIABLES INITIALIZED IN SUBROUTINES
C
      COMMON /INITV/SYS,TYPE,KP,ETAD,W1,K,CW,PW,KPC,
     & SPCHE,IMP,SEXU,SEXH,EFPHI,EFPUI,FUI,BSFCU,BSFCH,
     & SPCMG,SPCGEN,SPCCNT,SPCTRN,DPKWHE,ETAM,HBFI,SWENG,NYL,
     & SWTRN,AWC,GPO,MPGU,MPGCB,CHEF,MPGH,IPTO,NSY,NBC,IPSU,
     & ROEPS,ROSPP,ROPPS,ROEPSL,ROPLL,ROES,RHOSS,BDPKWH,CP,CL,YLI,EE,
     & FHE,FDE,FSS,YMI,DP,NSYS,NBCS,TYPESE,SYSSE,RI,GF,GK,CM,
     & DR,IR,NS,NF,TX,KPR,SWCVT,SPCCVT,SWCNT,SWMG,SWGEN
     & ,EMEDI,ETAPP,FCON,IF,NSCV,VICF,NLDF,ST,DMUP,CCV
     & ,MIFHV,LRC,ICF,MCMO,FINC,UDCC,BSV
C
      COMMON /INITM/SEX,WVO,ETAEU,ETACH,WDTO,GAMMA,YMP,ALPHA
```

```
14070       COMMON /INITS/CDTD,DTMG,SEXHCV,OPCTG,TACCV,AGCCV,EFPU,EFPH,FU,YM
14080C
14090       DATA ROEPS /18.,27.,30.,30.,50./
14100       DATA ROSPP /45.,65.,85.,45.,65./
14110       DATA ROPPS /55.,75.,95.,50.,80./
14120       DATA ROEPSL /20.,30.,33.,33.,55./
14130       DATA ROPLL /45.,65.,85.,45.,65./
14140       DATA ROES /0.,0.,0.,0.,0.,7./
14150       DATA RHOSS /150.,200.,150.,60.,80./
14160       DATA BDPKWH /50.,50.,60.,60.,40.,400./
14170       DATA CP /0.95,1.45,1.8,1.8,2.1,2.8/
14180       DATA CL /800.,800.,500.,1500.,800./
14190       DATA YLI /5.,5.,3.,6.,5.,10./
14200       DATA EE /.26,.26,.15,.18,.20/
14210       DATA BSV /5*.10/
14220       DATA MIPHV /.25,.25,.25,.25,.25,.25/
14230       DATA RI /75.,75.,35.,35.,35.,35./
14240       DATA FHL /0.,0.,.33,.60,.60,.33/
14250       DATA FDL /1.,1.,1.,.4,.4,1./
14260       DATA FSS /6*0./
14270       DATA CW,PW,KP,W1 /2150.,300.,0.028,2450./
14280       DATA ETAD,ETAPP /.80,.35/
14290       DATA AWC,K,YRI,FUI,FMEDI,SEXU,SEXH /.62,.3,11852.,.65,.75,.12,.127/
14300       DATA EFPUI,EFPHI,DPKWHE,GPO /.25,.90,.042,1.0/
14310       DATA SPCMG,SPCGEN,SPCCNT /20.,8.,6.7/
14320       DATA HPFI,CHEF,BSFCU,BSFCH,TMP /1.2,.75,.55,.60,30./
14330       DATA DR,IR,NS,NF,TX /.09,.12,12,4,.25/
14340       DATA IF,ST,DMUP,NLDF /.07,.05,.3,0./
14350       DATA KPC,SWENG,SWTRN /.028,6.,1.2/
14360       DATA SPCHE,SPCTRN,MPGU /10.,2.5,22./
14370       DATA MPGH,MPGCB,CCV,NSCV /32.,26.,5700.,10./
14380       DATA MCMO,LRC,FINC,ICF /2.,35.,125.,.01/
14390       DATA ETAM,VICF,ECON /0.9,.05,1/
14400       DATA NBCS,NBC,NSYS,NSY /1,5,1,6/



14410       DATA SWCNT,SWMG,SWGEN,UBCC /1.5,10.,4.,14./
14420       DATA CM,NYL /2.,1/
14430       DATA SYSSE,TYPESE /6,1/
14440       DATA IPTO,IPSU /1,1/
14450       DATA SWCVT,SPCCVT /2.25,3.3/
14460       DATA GF,GK,KPR,DP /.7,.011,.03,.8/
14470       END


$           NOTE    FILENAME LDSDATA
$           NOTE    LABELS- ASIS,R(LB)
$           NOTE    TABS/SETTNGS-  ASIS
```

Section 3
HYBRID VEHICLE SIMULATION PROGRAM (HYVEC)

Section 3

HYBRID VEHICLE SIMULATION PROGRAM (HYVEC)

3.1 HYVEC DESCRIPTION

This section contains a description and listing of the computer program that was used during Phase I to simulate vehicle operation.

The computer program (HYVEC) was developed to simulate second-by-second operation of the hybrid vehicle over urban and highway driving cycles. The program was used extensively in the Design Trade-Off Studies (Task 2) to evaluate in depth the hybrid power train configurations which were identified as the most promising in the first screening. HYVEC was also used in the Preliminary Design Task (Task 3) to update the hybrid vehicle energy-use and performance using refined component characteristics and vehicle weight projections. A complete listing of the program is given in Section 3.2.

A schematic of the HYVEC calculation procedure is shown in Figure 3.1-1. As indicated in Figure 3.1-1, the calculation for a particular driving cycle is performed starting at the wheels and working from component-to-component through the power train until the fuel and/or electricity needed to drive the vehicle for each increment of time is determined.

Detailed models based on experimental data and analysis are used for each of the power train components. For the electric drive system, motor voltage and current are determined and used as inputs to a battery model which describes the battery in terms of terminal voltage as a function of battery current and state-of-charge. Battery state-of-charge is expressed as the ratio of the Ah-used to the cell Ah capacity at the time-averaged discharge current. All the electrical power train components are modeled using scaling factors which permit the component sizes (ratings) to be changed without altering the basic inputs to the program. The electric motor is described in terms of its continuous rated power, base speed, and nominal rated voltage and flux. The battery is described in terms of cell Ah-rating at the C/3 rate and the number of cells in each battery module (i.e., nominal battery voltage).

The mechanical driveline components, the heat engine and transmission, are modeled in a conventional manner. The heat engine is described by its maximum power and rpm. Fuel consumption and emissions characteristics are input as maps of bsfc and bsem (brake specific emissions = HC, Co, No_x, particulates) as functions of percent speed and percent of the maximum power at that speed fraction. The multispeed gearbox transmissions are described in terms

of the gear ratio and efficiency in the various gears and the pumping losses if the gearbox is hydraulically shifted. The steel-belt CVT is described in terms of the maximum reduction speed ratio and the maximum overdrive speed ratio. Friction and pumping losses are combined into a single, speed dependent loss term for the CVT.

The control strategy for operating the hybrid power train is described in HYVEC by a series of statements which specify under what conditions the engine is on, what fraction of the power required is supplied by the electric motor, when the gearbox should be shifted or the battery charged, how the accessory loads should be met, etc. Development of the control strategy for the hybrid vehicle was a key part of the Phase I study and the HYVEC program was an important tool in that development.

The HYVEC program was also used to calculate the maximum effort acceleration performance of the hybrid vehicle. In that calculation, both the heat engine and electric motor are operated at the maximum power (or torque) attainable from them at each vehicle speed. The gear shifting strategy is such that the motor and engine are permitted to operate much nearer their maximum rpm than in usual driving. Particularly for the heat engine, this increases the power available at moderate vehicle speeds. The maximum power attainable from the electric drive system depends on the state-of-charge of the battery. As the battery charge is depleted, the voltage droop of the battery increases at high currents and the maximum power the battery can provide becomes smaller. Maximum effort acceleration calculations at specified levels of battery state-of-charge can be made with HYVEC.

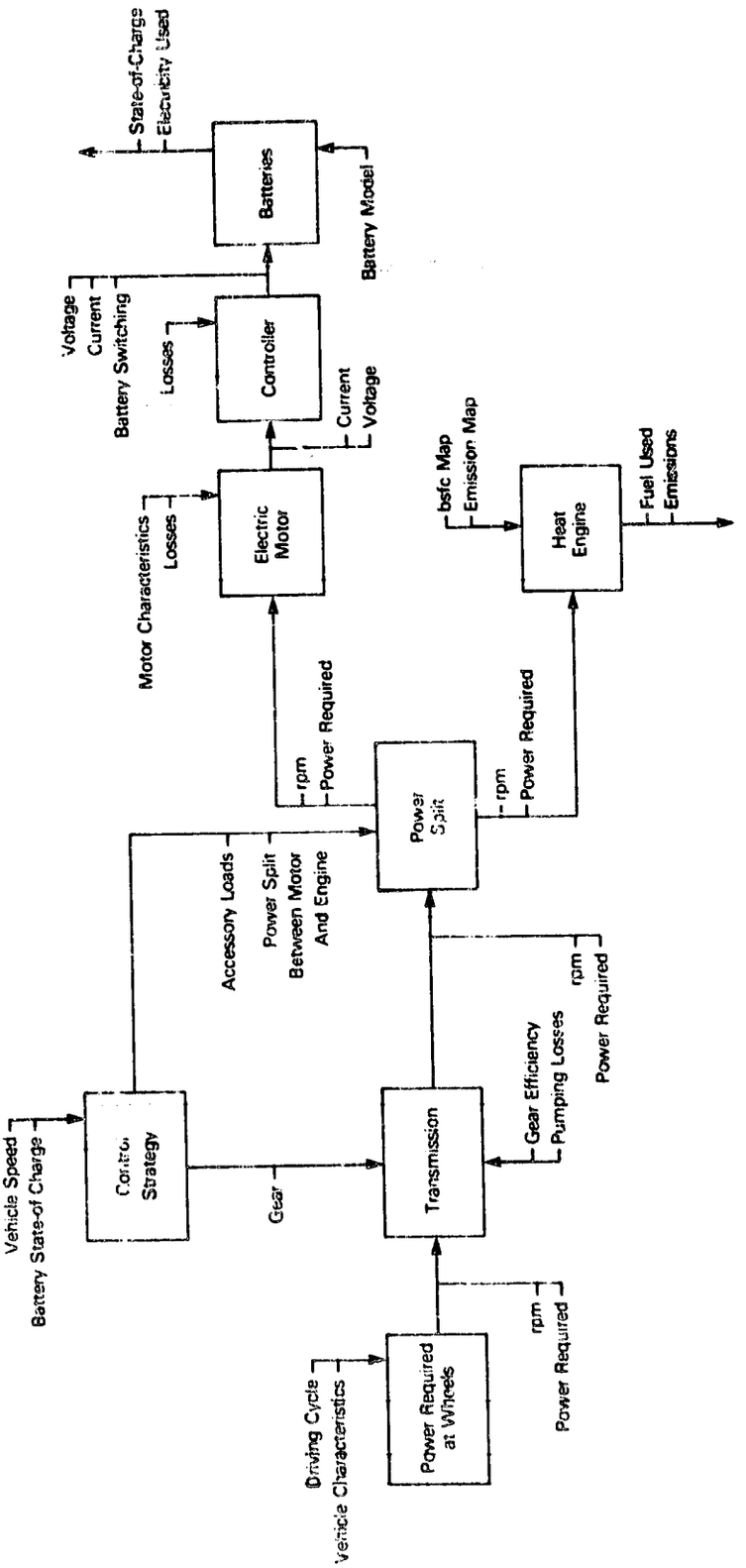


Figure 3.1-1. Schematic of the Hybrid Vehicle Simulation Calculation (HYVEC)

3.2 HYVEC LISTING

```
10C     ****************************************************
20C     ********** HYBRID VEHICLE PERFORMANCE SIMULATION PROGRAM ************
50C     ****************************************************
60C     INCLUDE 'HYPR1'
70C     INCLUDE 'HYPR2'
100      NAMELIST /IN/S,CD,AF,ND,IFIL2,BCMN,WFMX,IFIL,CFEF,CAEF,TSS,
110     &    WMMX,DT,DWHL,REGEN,NCYC,MTYP,VEON,VBCMN,MV,VWIND,CHGREF,
120     &    BCMX,PRCMX,GTYP,TTYP,DPRT,BCI,IPRTS,WIDLEM,CROL1,IESIZ,
130     &     CTYP,GRT,ETYP,SSVEL,IDOWN,DCTYP,DIST,VMODE,NTMTR,BCEFF,
140     &      KP,PFL,PFM,PFFW,MFB,ESP,TSP,MSP,KOL,CROL,CROL2,IACCEL,IFIL3,
150     &      CSP,TFWP,FWSE,MFPT,MCHN,MPL,KMP,KTP,WFWMX,ATEMP,RHOF,VMIN1,
160     &     ACTYP,IFAN,IPS,IALT,IAC,IWP,DTTYP,PRAT,IESF,JWHL,ITRIP,
170     &     FIDLE,WIDLE,FSTRT,SHIFT,SHIFTA,GRATS,NTM,JENG,JMTR,WIDLEA,
180     &     EPCP,IECON,BCVAL,IINIT,BSEC,BSPC,FDCMXR,BCMXR,BSPN,BSEN
190     & ,PFMX,SPFR,EFONOX,EFOHC,EFOCO,CTCT,TCHWO,ETON,ETOF,SPFRA
200     &,IAUF,IAON,ITCD,IVAC,TSENG,WSF,QHEAT,QCONST,IEAT,IHD,PTLOSS,
210     &CWK,CFF,IEM,PFLOSS,SNFW,RSS,PHEMAX,JACCEL,CCC,SACC
211      &,ICVT,ECVT,WFACT,ODR,RR,PFMIN,PFMAX
220      NAMELIST/INZ/EXPO,CKBATT,VO,XR,XS,RR,RS,
230     &BK3,BK1,BK2,BKFV,BKW,ENW,ENF,BKF,ENC
240     &EMC1,EMC2,WDAC,PHIAC,VOLTAC,VATAC
250C
260      NAMELIST /INY/NF,NW,WWF1,WWDG1,RRA,RRF,RRL,AKT,AKV,WDAT,AFCMAX
270     &,RST1,RST2,AACMAX,BBD1,BBD2,BBD3,AFFO,AFMIN,TOLR,FEFF,IALB,RCNF
280     &,WWCP,JELEC,ITR,SS,EII,MODE,ARTURN,FTURN,PLOGIC,PMCON,RCNZ,PB
290     &CON,PCHRG,DMOTI,NC,WBASE,VBASE,FBASE,VPHI,VWCL,VAT,ALPHA,BETA
300     &,VBASE1,ITMAX,WBASE1,IBASE1,IBASE,NS,NP,FBASE1,UAHC,FLUXI,EFMOTO
310     &,CURLOW,EQVFH,NS2,NP2,UAHC2,IBTYP,NC1,NC2,IBS,IBTYP,VCNZ,VCNF
320        REAL YTEMP(20)
330C     ***************** INITIAL SECTION ************
340C
350      DATA TTYP/1/BTYP/1/NCYC/1/VN/0./VO/0./SSVEL/2*50./
360      DATA IKNT/1/ITRIP/25*0/ITRDC/25*0/TAC/0./ITRNUM/25*0/I1/0/TSS/2*10./
370      DATA DPRT/5./BCMX/.85/BCMN/.2/KM/0./IFIL/0/EM/.7/ND/.96/IA1/'1'/
380      DATA S/0./DWHL/.5/DT/1./WFMX/5000./WMMX/5000./IESF/0/EON/1/
390      DATA CTYP/1/ETYP/1/DCTYP/4/MTYP/1/VMODE/20./IA2/'0'/IINIT/1/
400      DATA IDOWN/0/REGEN/0/FUEL/0./A/0./GRM/1.4/IEIF/0/WFWMX/0./INS/0/
410      DATA NTM/.95,.96,.97,.98/SHIFT/20.,40.,80./SHIFTA/40.,70.,110./
420      DATA VEON/20./IT1/0/VBCMN/10./IBCF/0/MB/0./IFIL2/0/CROL1/1.4E-3/
430      DATA IC/1/ICYC/1/T/0./DIST/0./ERGEN/0./DTTYP/1/PRAT/0./IACCEL/1/
440      DATA IRRUN/0/TO/1.E10/TMEFF/0./TEENG/0./IMON/0/IEON/0/JENG/.07/
450      DATA ICNNT/3*1/GRATS/3.1,2.5,1.5,1.,2.75/NTMTR/.98/JMTR/.07/
460C     ************** VSIZ ROUTINE DATA **************
470      DATA KP/.05/PFE/.5/PFM/.5/PFFW/0./MFB/.2/ESP/.391/TSP/2.286/
480      DATA MSP/.25/KOL/3./CSP/.641/BSE/0./BSP/0./TFWP/20./FWSE/.1/
490      DATA MFPT/.5/MCHN/500./MPL/150./KMP/.22/KTP/1./C/3.6/MV/0./
500C     RERC DATA
510      DATA CD/.45/AF/2./EVD/0./CROL/.012/CROL2/1.2E-5/JWHL/1.4/
520C     MOT/GEN DATA
530      DATA CFEF/.99/CAEF/.96/VWIND/0./WIDLEM/600./IESIZ/0/
540C     BAT DATA
550      DATA BCI/1./BTYP/1/EBOUT/0./EBIN/0./IECON/0/EPCP/1./
560      DATA BSEC/6*2.E-2/BSPC/6*4.41E-3/IA3/'  SPEED'/IA4/'  POWER'/
570      DATA BSEN/6*3./BSPN/6*1./
580      DATA BDCMXR/6*4./BCMXR/6*1./
590C     ACC DATA
600      DATA ACTYP/2/IFAN/1/IPS/1/IALT/1/IAC/0/IWP/1/
610C     ENGINE
620      DATA FSTRT/10*0./WIDLE/8*800.,900.,750./WIDLEA/1200./
630      DATA FIDLE/2.45,1.32,4.9,2.6,2.7,.7,1.6,3.6,2.42,1.2/
```

```
      DATA ISTOP/0/,IFIL3/0/,ATEMP/70./,CHGEFF/1./,RCEFF/1./,BCVAL/.5/
C
C     ****************** DATA INPUT ********************
C
      CALL FILES
       DO 77 I=1,20
  77    YTEMP(I)=0.0
C$$$$$$ WANT TO READ FROM DEVICE 15&16 FIRST TIME THROUGH THE PROGRAM
      READ(15,IN)
       READ(17,INY)
          READ(19,IN2)
      IRUN2=0
      GO TO 16
  14  IRUN2=0
       WRITE(6,98)
  98   FORMAT('DO YOU WANT TO CHANGE  INPUT OTHER THAN MOTOR ?,TYPE YES
     & OR NO')
       READ(5,101)MOS
       IF(MOS.EQ.'NO') GO TO 99
      WRITE(6,15)
  15  FORMAT(5X,'INPUT $IN:')
      READ(5,IN)
  99   WRITE(6,100)
 100   FORMAT('DO YOU WANT TO CHANGE MOTOR DATA?, TYPE YES OR NO')
       READ(5,101)MOS
 101   FORMAT(A2)
       IF(MOS.EQ.'NO') GO TO 16
       WRITE(6,102)
 102   FORMAT(5X,'INPUT $INY:')
       READ(5,INY)
  16  CONTINUE
        TSAV=ETOP
        IF(IFIL.NE.0) CALL ASSIGN(21)
        IF(IFIL2.NE.0) CALL ASSIGN(22)
        IF(IFIL3.NE.0) CALL ASSIGN(23)
        CALL ASSIGN(18)
       WIDE=WIDLE(ETYP)
        WIDLEM=WBASE
        IF(IBS*JELEC.EQ.2) WIDLEM=WBASE/2.
         IF(JELEC.EQ.1) WIDLEM=600.
       WIDM=WIDLEM
       IF(SHIFTA(1).EQ.0.)CALL LIN3(0.,1.,1.,1.)
         ITR=1
       PRT=0.
       DTP=DT
       IF(ITRIP(1).LT.2)GO TO 25
       I=ITRIP(1)
       DO 17 I1=1,I
       PEMX=0.1*FLOAT(ITRIP(I1+1))
       ITRNUM(I1)=IFIX(PEMX)
       ITRDC(I1)=ITRIP(I1+1)-ITRNUM(I1)*10
  17   CONTINUE
       I1=0
       DCTYP=ITRDC(1)
  25  CALL VSIZ
C     CHECK THESE NUMBERS
       PBDCMX=BDCMXR(BTYP)*PSBMX
       PBCMX=BCMXR(BTYP)*PSBMX
       BCHG=BCI
       HW=BCHG*MB
```

```
      IF(IACCEL.EQ.2)GO TO 88
      GO TO 90
   88 CALL OUTPUT
      GO TO 91
   89 CALL VSIZ
      BW=BCHG*MD
C
C     ********************* MAIN SECTION ******************
C
   90 CONTINUE
      IRRUN=0
C     CALS: VEL*ACCEL (KM/HR)
      CALL DCYCLE
  111 CALL DSPWR
      CALL TRANS(1)
  222 CALL COMPAR
      CALL CNTL
       PMSAVE =PM
C
C       HEAT ENGINE STARTING WITH MOTOR
        JSTART=0
        IF(EONSAV.EQ.1) GO TO 200
        IF(PM.GE.PHEMAX.AND.VNEXT.GT.V) GOTO 250
        IF(VNEXT.GT.VMOD)GO TO 250
        GO TO 200
  250   JSTART=1
        PM=PM+PSTART
  200   EONSAV=EON
        MONSAV=MON
C
C         COMPUTE ENGINE ON/OFF TIME
C
        IF(EON.EQ.0) GO TO 777
        ETON=ETOF+DTP
        ETOF=0.0
        GO TO 707
  777   ETON=0.0
        ETOF=ETOF+DTP
        TSAV=ETOF
  707   CALL ENG(2)
  555   CALL MTR(2)
        ITR=ITR+1
      PE=PC
  333   IF(CTYP.NE.3) CALL BAT(2)
          SNEW=55
C
C       STORE DATA FOR PLOTTING
          IF(WE.EQ.0.0)WE=0.01
          IF(WM.EQ.0.0)WM=0.01
        TTEN=PE/WE
        TMOT=PM/WM
        TT=FLOAT(ITER)
        IF(ITER.GT.2000) GO TO 666
  444   WRITE(13,515)TT,PM,WM,IARM*IBASE,VCHOP*VBASE,IBAT2*VBASE,
     &PBAT2*PBASE,ABAT2*IBASE,ECHG,PE
     &  ,WE,PDS,EBOUT,FLOX,TTEN,TMOT,A,FUEL,V,EMNOX
  515   FORMAT(2(10E12.5/))
C/
  666   CALL OUTPUT
        IF(IRRUN.EQ.1) GO TO 89
```

```
1830   91 IF(.RUN2.EQ.1) GO TO 14
1840      IF(.STOP.NE.1) GO TO 90
1850C$$$$$$ WANT TO DETACH DEVICE(FILE) 15 FROM AFT
1860      CALL FILES
1870      STOP
1880      END
```

```
10C    HYPR1 PROC
20              PARAMETER IOW=200,IOD=17
30              IMPLICIT REAL(J-N)
40C  SET UP SEVERAL VARIABLES AS CHARACTER STRINGS
50         CHARACTER*8 IFORM2,IA3,IA4,JDATE,JTIME
60              INTEGER REGEN,DCTYP,CTYP,MTYP,ETYP,TTYP,DTTYP,
70        &        FTYP,NCYC,ION,MM,MON,ACTYP
80              DIMENSION ITRIP(25),ITRNUM(25),ITRDC(25),SSVEL(2),TSS(2),IFORM2(6)
90              DIMENSION BSEC(6),BSPC(6),NTM(4),FIDLE(10),FSTRT(10),WIDLE(10)
100              DIMENSION OAR(10D,IOW),ICNNT(3),SHIFT(3),SHIFTA(3),GRATS(5)
110              DIMENSION PACC(4),BSPN(6),BSEN(6),BDCMXR(6),BCMXR(6)
112               DIMENSION PHIAC(10),VOLTAC(10),VATAC(10)
120C
130C     ****** ORIGINAL COMMON-REAL PARAMETERS ** MAY 15
140C
150         COMMON /REAL1/ WM,
160        &     PM,PC,PB,PEMX,V4,V5,V6,BSEC,PT,MV,NTMT
170        &     ,EM,RCHG,PJM,A,BCMX,BCMN,PBCMX,PJE,TFIN,DTP,VMODE,NTM,WIDM
180        &     ,PSBMX,ESBMX,PFWMX,TMMX,PMMX,WF,PF,PMXM,PMXE,WDS,WIDE
190        &     ,V,NT,ND,VBCMN,WT,DWHL,PWHL,GRATS,PDS,GRT,CROL1
200        &     ,EBOOT,EBIN,EVD,ERGEN,BCI,FUEL,PACC,PRAT,CROL,CROL2,JWHL
210        &     ,PBDCMX,MCHN,EENG,TEENG,MVCH,TMEFF,ESP,TSP,MSP,BSE,BSP
220        &     ,EBAT,ECONS,OAR,NTMTK,JE,JN,PEEA,PEMA,VWIND,WFWMX,TAC,WIDLEM
230        &     ,JMTR,JENG,S,WIDLEA,SHIFT,SHIFTA,K1,K2,K3,K4,TSS,SSVEL,
240        &   AF,BSEC,BSPC,BW,C,CAFF,CD,CFLF,CSP,DIST,DPRT,RHOF,VMIN1,SPERA,
250        &    FIDLE,FSTRT,FWSE,GRM,KM,KMP,KOL,KP,KTP,MB,WEMX,
260        &    MEB,MEPT,MPL,PEE,PEEW,PEM,PNI,WEH,TFWP,WIDLE,WMMX
270C
280C     ****** NEW COMMON-REAL PARAMETERS **
290C
300         COMMON /REAL2/VEON,ATEMP,BCLEF,CHGEFF,EPCP,BCVAL,PRT,
310        &BSPN,BSEN,BDCMXR,BCMXR,EMHC,EMCO,EMNOX,EFOHC,EFOCO,EFONOX,PEMX,SPER
312        &,CTCT,ETON,ETOF,TCHWO,EGMCT,EGMCOT,EGMNOXT,EGMCST,EMCS,TSAV
313        &,IAOF,IAON,ITCD,PVLOSS,TSPNG,WSE,PSTART,BURNM,QHEAT,QCONST,
314        &IEAT,IDD,PTLOSS,PFLOSS,IEM,IVAC,CCC,SACC,CWW,CFF
315        &,IFIRST,JSTART,VNEXT,ANEXT,PHEMAX,JACCEL,PSS,SNEW,VMOD
316        &,ICVT,ECVT,WFACT,ODR,RR,PEMIN,PEMAX
320C
330C     ****** ORIGINAL INTEGER BLOCK= MAY 15 *****
340C
350         COMMON /IREAL1/REGEN,
360        &    DCTYP,CTYP,MTYP,ETYP,TTYP,ITER,IMGF,DTTYP,IRUN2
370        &   ,JDATE,JTIME,BTYP,NCYC,ION,MM,IFIF,IRRUN,IESTZ
380        &   ,IFIL,IFAN,IPS,IALT,IAC,IWP,IBCF,IESF,IACCEL,IGEAR
390        &   ,MON,ACTYP,IPRTS,IEOF,IMON,ICNNT,I1,IDOWN,IFIL2,III
400        &   ,ITRIP,ITRNUM,ITRDC,IKGT,IFORM,IFORM2,IA1,IA2,IA3,IA4,IC
410C
420         COMMON/MOTDAT/DMOTI,WBASE,VBASE,EBASE,IBASE,RBASE,PBASE,FLUXI
430         COMMON/PCDATA/NF,N,NC,WWF1,WWPG1,FRA,RRF,RRL,AACMAX,WCP,JFLUX,
440        &AFCMAX,AKV,AKT,WDAT,RST1,ST2,BRD1,BRD2,BRD3,AFEO,WBASE1,EBASE1,
450        &AFMIN,TOLR,FEFF,IALG,WWCP,ARTURN,FTURF,IBASE1,PBASE1,VBASE1
460         COMMON/COMDAT/JELEC,ITR,ITMAX,VWCL,VPHI,VAT,CURLOW,JTORQ,IAPM,
470        &MODE,PLOGIC,ECHRG,PBCOM,PSCON,RPSAVE,FLUX,VCHOP
490         REAL IBASE,IARM,IBATT,IELF,NW,NF,NC,IALR,KE,KT,ILOSS,IBASE1,
491        &NS2,NP2,NC1,NC2,
500        &NP,NS,ALPHA(5),BETA(5),VPHI(10),VWCL(10),VAT(10),BETAT(5),ALPHAT(5)
510         COMMON/BATDAT/NS,NP,E1,E0,E2,ALPHA,BETA,ECO,EII,QAHC,ENBAT,EFMGTO,
520        & EIBATP,ENSATP,SS,IBATT,PBATT,EBATT,EETAT,ALPHAT,EIT,NC1,NC2
521        &,EBVEM,NS2,NP2,APBAT2,QAHC2,AHC2,DCRATE,AH02,ACOR,EBAT2,IBTYP
522        &,PBAT2,IDS,VCN2,VCNF,RCN2,RCNF,EBMET
```

```
530C     ***** NEW INTEGER BLOCK ********
540C
550      COMMON /IBLK2/ISTOP,IFIL3,IINIT,IECON
560C     END
570C     HYPR2 PROC
580           COMMON /RBLK3/PE,WE,DT,V0,VN,T0,T,XF,YF,V1,V2,V3
590           COMMON /IBLK3/ITI,I,IX,II,ISUB,ICYC,IY
600C     END
```

```
10C
20C     ************** CONTROL SUBROUTINE ******************
30C
40      SUBROUTINE CNTL
50C     INCLUDE HYPR1
60C     INCLUDE HYPR2
100      ISUB=1
110        DIMENSION PCVT9(12),PCVT10(12),SPCVT9(12),SPCVT10(12)
120        DATA PCVT9/0.,16.7,22.6,32.9,39.4,46.,52.5,59.3,66.,75.7,
130     686.,100./
140        DATA SPCVT9/20.,20.,26.6,33.3,40.,47.,53.,60.,67.,76.7,86.7
150       6.,100./
160        DATA PCVT10/0.,8.6,11.5,14.4,17.3,23.,38.1,50.,61.3,75.7
170       6.,78.4,100./
180        DATA SPCVT10/30.,30.,30.,30.,30.,40.,40.,50.,60.,80.,90.
190       6.,100./
200          PWAT=PRAT
210          IGO=0
220        IF(JVMIN.EQ.0) GO TO 30
230          IF(ABAT2.GT.IBASE/2.) GO TO 40
240           JVMIN=0
250           GO TO 40
260  30     EMIN1=VMIN1*NC2*NS2/(VBASE)
270         EMIN2=1.3*AFMIN*WM*AKV/VBASE
280         VMIN=(AMAX1(EMIN1,EMIN2))/NP
290         IF(EBAT2.GT.VMIN) GO TO 40
300         JVMIN=1
310  40     PFFX=PFMX*(1.0-.5*FLOAT(JVMIN))
320  50          OAR(3,192)=0.0
330          VMOD=VMODE
340         IF(SS.GT.0.5) VMOD=VMODE*(1.0-SS**2.)
350C
360         IF(ICVT.EQ.1) GO TO 8000
370        GO TO (100,1000,2000,3000,4000,5000,6000),CTYP
380C
390C       ********* CONTROL STRATEGY  #1 ********
400  100   IF(PT+PJM)300,,104
410        IF(V)360,360,
420  104   IF(IBCF),,120
430  105   IF(BCHG-BCMN)130,130
440  115   IBCF=1
450        MON=1
460        EON=1
470C       ****** BATTERY NEEDS CHRG, FIRST CHECK FOR PWR FROM HE
480        IF(PMXE*EPCP-(PT+PJE+PJM+PACC(1)))150,160,
490C       ********** HENG HAS EXTRA POWER TO CHG BATT ***********
500        PE=PMXE*EPCP
510        PM=PMXE*EPCP-(PT+PJE+PJM+PACC(1))
520        IF(PM.GT.PMXM)PM=PMXM
530        PE=PM+(PT+PJE+PJM+PACC(1))
540        IMGF=1
550        GO TO 400
560C       ************ HENG HAS JUST ENUF POWER FOR DSHAFT ******
570  160   PE=PT+PJE+PACC(1)
580        MON=0
590       GO TO 400
600C       ******* BCHG IS POSITIVE- CHECK FOR BCMX ******
610  120   IBCF=0
620        IF(BCHG-BCMX)115,115,105
630C       ***** BAT DOESNT NEED CHG, PT IS POS, DIST PM-PE
```

```
640C      CNTLDT MODIFICATIONS
650C      FOR COMMON PFFX
660C       SET PFFX IN NAMELIST=0.8
670    130 PM=PT+PACC(4) +PJM
680         MON=1
690         PFFX=0.80
700         IF(IROE)21,21,22
710      21 IF((PMXM*(1.0-.5*55**2))*.5-PM) 23,23,410
720      23 IROE=1
730      22 IF(PMXM*(1.0-.5*55**2)*PFFX-PM)25,25,410
740      25 LON=1
750      26 V1=PMXM*(1.0-0.5*55**2)*PFFX
760         IF(PWAT)27,27,28
770      27 PE=PT+PJM+PJE+PACC(1)-V1
780         PM=V1
790         GO TO 29
800      28 PM=(PT+PACC(1)+PJM+PJE)*(1.0-PWAT)
810         PE=PM*PWAT/(1.0-PWAT)
820C
830      29 IF(PM.GT.PMXM.OR.PE.GT.PMXE)GOTO 140
840         GO TO 400
850C
860 140   IF(IGO.EQ.1) GO TO 141
870         PWAT=PMXE/(PMXE+PMXM)
880         IGO=1
890         IF(CTYP.EG.5) GO TO 28
900 141 IF(OAR(3,192)),,144
910 143   CALL TRANS(3)
920           PWAT=PMXE/(PMXE+PMXM)
930       OAR(3,192)= 1.
940           IGO=0
950       GO TO 130
960 144 OAR(3,192)=0.
970           IF(IGEAR.NE.1) GO TO 143
980       GO TO 400
990C      ****** BAT NEEDS CHG BUT HE DOES NOT HAVE PWR TO MEET DCYCLE R
1000 150 IF(PMXE*EPCP-(PT+PJE+PACC(1))),160,160
1010       PE=PMXE*EPCP
1020       PM=PT+PJM+PACC(1)+PJE-PE
1030       IF(PM.GT.PMXM)PM=PMXM
1040       PE=PT+PJM+PACC(1)+PJE-PM
1050       IF(PMXE-PE),400,400
1060       IF(OAR(3,192)),,144
1070       CALL TRANS(3)
1080       OAR(3,192)=1.
1090       GO TO 150
1100C      ***** POWER AT MTR IS NEGATIVE ******
1110 300 IF((PT+PJM)+PACC(4)),400,330
1120       IF(REGEN)340,340,
1130         IF(BCHG.GE.0.9) GO TO 340
1140       IF(V-VBCMN)340,,
1150       IF(BCHG-1.),340,340
1160       PM=ABS(PT+PJM+PACC(4))
1170       MON=1
1180       IMGF=1
1190       IF(PM.GT.PMXM)PM=PMXM
1200       ERGEN=ERGEN+PM*DTP/3600.
1210       OAR(2,192)=OAR(2,192)+((PT+PJM+PACC(4))+PM)*DTP/3600.
1220       GO TO 400
1230C      *****RGEN PWR INSUFFICIENT TO DRIVE ACC **
```

```
1240  330 PM=PACC(4)+(PT+PJM)
1250      MON=1
1260      GO TO 400
1270C     *** ENUF PWR TO DRIVE ACCS BUT NO REGEN WANTED **
1280  340 DAR(2,192)=DAR(2,192)+(PDS+PJM+PACC(4))*DTP/3600.
1290C     ABOVE IS BRAKE URG
1300        WXX=GRM*WT
1310        IF(IEM.NE.0.AND.DCTYP.NE.7)WM=AMAX1(WXX,WIDLEM)
1320      GO TO 400
1330C     ***** VEHICLE VEL IS ZERO ******
1340  360 IF(IBCF)380,380,
1350C     ARBITRARY SPEED ASSIGNMT FOR BAT CHRGNG
1360      WE=.5*WEMX
1370      EON=1
1380      CALL ENG(1)
1390      CALL ACC
1400C     ARBITRARY PWR ASSIGNMNT
1410      PE=.6*PMXE
1420      MON=1
1430      PM=PE-PACC(1)
1440      WM=WE
1450      IMGF=1
1460      IF(PM.GT.PMXM)PM=PMXM
1470      PE=PM+PACC(1)
1480      GO TO 400
1490C     ***** V=0 MTR SUPPLIES ACC PWR
1500  380 IF(IECON),,390
1510      MON=1
1520C     MAY BE TOO SLOW
1530      WM=WIDM
1540      CALL MTR(1)
1550      CALL ACC
1560      PM=PACC(4)
1570      GO TO 400
1580C     ****** V=0 ENG SUPPLIES ACC PWR
1590  390 EON=1
1600      WE=WIDE
1610      PE=PACC(1)
1620  400 IF(EON),,410
1630      IF(IECON)410,410,
1640      EON=1
1650      IEIF=1
1660  410 RETURN
1670C     ****** CONTROL #2 ********
1680 1000 IF(PT+PJE)300,,1104
1690      IF(V)360,360,
1700 1104 IF(IBCF),,1120
1710 1105 IF(BCHG-BCMN)115,1130,1130
1720 1120 IBCF=0
1730      IF(BCHG-BCMX)115,115,1105
1740C     *** BAT DOESN'T NEED CHARGE-----PDS IS POS---DIST PWR
1750 1130 PE=PT+PJE+PACC(1)
1760        IF(IEM.NE.0.AND.DCTYP.NE.7) PE=PE+PJM+PFLOSS
1770        IF(IEM.NE.0.AND.DCTYP.NE.7) WM=WE
1780      EON=1
1790      IF(PMXE*EPCP-PE),400,400
1800 1132 V1=PMXE*EPCP
1810      MON=1
1820      IF(PWAT),,1134
1830      PE=V1
```

```
      PM=PT-(V1-PJE-PACC(1))*PJM
      PM=AMIN1(PM,PMXM)
      PE=PT-(PM-PJM)*PJE*PACC(1)
      GO TO 1138
1134  PE=(PT*PACC(1)*PJM*PJE)*PWAT
      PM=PE*(1.-PWAT)/PWAT
1138  IF(PE.GT.PMXE.OR.PM.GT.PMXM)GO TO 1140
      GO TO 400
1140    IF(IGO.EQ.1) GO TO 1141
       PWAT=PMXE/(PMXE+PMXM)
        IGO=1
        IF(CTYP.EQ.5) GO TO 1134
1141  IF(OAR(3,192))*,1144
1143   CALL TRANS(3)
        PWAT=PMXE/(PMXE+PMXM)
      OAR(3,192)=1.
      GO TO 1130
1144  OAR(3,192)=0.
        IF(IGEAR.NE.1) GO TO 1143
      GO TO 400
C     ********** CONTROL #3 **********
2000  EON=1
      IF(V)2100,2100,
      IF(PT*PACC(1)*PJE)2100,2100,
      PE=PT*PACC(1)*PJE
      IF(PE-PMXE)2400,2400,
      IF(IESIZ)*,2200
      DO 2010 I4=1,3
      CALL TRANS(3)
      MON=0
      PE=PT*PACC(1)*PJE
      IF(PE-PMXE)2400,2400,
2010  CONTINUE
      RETURN
2100  IEIF=1
C     BRAKE NRG
      OAR(2,192)=OAR(2,192)+(PT*PACC(1)*PJE)*DTP/3600.
      RETURN
2200  IRRUN=1
      WRITE(6,2250)T,V,A,PT,PACC(1),PMXE,WE,KP
2250  FORMAT(1H,'KP INCREASE: T,V,A,PT,PACC,PMXE,WE,KP',/
     &       8E10.4)
      KP=KP+.007
2400  RETURN
C     ******* CONTROL #4 ************
3000  MON=1
      IMGF=0
      PM=PT*PACC(4)*PJM
      IF(PM)3020,3020,
      IF(PM.GT.PMXM.AND.IESIZ.EQ.1)GO TO 2200
      IF(PM.LE.PMXM)RETURN
      DO 3010 I4=1,3
      CALL TRANS(3)
      EON=0
      PM=PT*PACC(4)*PJM
      IF(PM-PMXM)2400,2400,
3010  CONTINUE
      RETURN
3020  PM=0.
      MON=0
```

```
2440      GO TO 300
2450C     ***** CONTROL #5 ******
2460 4000 IF(V-VMOD)100,100,1000
2470C     ********** CONTROL #6 **********
2480 5000 IF(BCHG-BCVAL)1000,1000,100
2490C
2500C     ******* CONTROL # 7 **********
2510C
2520 6000 IF(BCHG-BCVAL)1000,1000,
2530      IF(V-VMOD)100,100,1000
2540C
2550C         ICVT=1 FOR CVT (CONTINOUSLY VARIABLE TRANSMISSION)
2560C
2570 8000    PACS=PACC(1)
2571        IF(WDS.EQ.0.0) WDS=0.00001
2580         IF(CTYP.EQ.3) GO TO 8200
2590         IF(CTYP.EQ.4) GO TO 8400
2591         IF(V.EQ.0.0.AND.IECON.EQ.1) GO TO 8200
2600         IF(VMOD-V)8200,8400,8400
2610C
2620C         HEAT ENGINE PRIMARY
2630C         CAN HEAT ENGINE ALONE PRODUCE POWER REQD. ?
2640 8200    PE=PT+PJE+PACC(1)
2650         IF(PE.LT.0.0) GO TO 8400
2660         IF(IEM.NE.0.AND.DCTYP.NE.7)WM=WE
2670         IF(IEM.NE.0.AND.DCTYP.NE.7) PE =PE+PJM+PFLOSS
2680         PEF=PE/PEMX*100.0
2690         IF(CTYP.EQ.3) GO TO 8205
2700         IF(PE.GT.PMXE) GO TO 8700
2710 8205    IF(PEF.GT.100.0) PEF=100.0
2720         IF(ETYP.EQ.10) GO TO 8210
2730  8209 CALL LIN2(PCVT9,SPCVT9,12,PEF,WEF)
2740           GO TO 8250
2750 8210     CALL LIN2(PCVT10,SPCVT10,12,PEF,WEF)
2760 8250     IF(WEF.GT.100.)WEF=100.
2770        WE=WEMX*WEF/100.
2780        GRT=WE/WDS
2790        IF(GRT.LT.1./ODR) GRT=1./ODR
2800        IF(GRT.GT.RR) GRT=RR
2810        WE=WDS*GRT
2820        EON=1
2830        MON=0
2840        GO TO 400
2850C
2860C       MOTOR PRIMAR
2870C         CAN MOTOR SUPPLY ALL THE POWER
2880C
2890 8400   PM=PT+PJM+PACC(4)
2900       GRT=WBASE*WFACT/WDS
2910        IF(GRT.GT.RR) GRT=RR
2920         IF(GRT.LT.1./ODR) GRT=1./ODR
2930       WM=WDS*GRT*GRM
2940        IF(PM.LT.0.0) GO TO 8500
2950         IF(WM.LT.WIDLEM) WM=WIDLEM
2960         EON=0
2970         MON=1
2980         IF(CTYP.EQ.4) GO TO 400
2990         IF(PM.GT.PMXM) GO TO 8700
3000         GO TO 400
3010 8500     PM=0.0
```

```
3020         IMGF=0
3030         MON=0
3040         EON=0
3050         IF(REGEN.EQ.0.OR.PCHG.GT.0.) GO TO 400
3060         PM=ABS(PT+PJM+PACC(4))
3061        GRT=RR
3062         WX=WDS*GRM*RR
3063         IF(WX.GT.WMMX) WX=WMMX*SPER
3064         GRT=WX/(GRM*WDS)
3065         IF(WX.LT.WIDLEM) WX=WIDLEM
3066         WM=WX
3070         IMGF=1
3080         MON=1
3081         GO TO 410
3090C
3100C          COMBINED MOTOR /ENGINE OPERATION
3110C
3120 8700     PTOT=PT+PJM+PJE+PACC(1)
3130         PART=PMXE/(PMXE+PMXM)
3140         PM=PTOT*(1.-PART)
3150         PE=PTOT*PART
3160         PEF=PE*100./PEMX
3161          MON=1
3162         EON=1
3170        IF(ETYP.EQ.10) GO TO 8410
3180         CALL LIN2(PCVT9,SPCVT9,12,PEF,WEF)
3190         GO TO 8420
3200  8410 CALL LIN2(PCVT10,SPCVT10,12,PEF,WEF)
3210 8420   IF(WEF.GT.100.0) WEF=100.0
3220         WE=WEF*WEMX/100.
3230         GR2=WE/WDS
3240         WM=WBASE*WFACT
3250         GR1=WM/WDS
3260 8750   GRT=AMAX1(GR1,GR2)
3270         IF(GRT.LT.1./ODR) GRT=1./ODR
3280         IF(GRT.GT.RR) GRT=RR
3290          WM=WDS*GRM*GRT
3300         WE=GRT*WDS
3310          IF(WM.LT.WIDLEM) WM=WIDLEM
3320          IF(WE.LT.WIDLE(ETYP)) WE=WIDLE(ETYP)
3330         WF=WE/WEMX*100.
3340C          CORRECT ACCESSORY POWER
3350 8800       CALL ACC
3360         DELPAC=PACC(1)-PACS
3370         PE=PE+DELPAC*PART
3380         PM=PM+DELPAC*(1.-PART)
3390         GO TO 410
3400       END
```

```
10C
20C     ********** VEHICLE SIZING ROUTINE **********
30C
40      SUBROUTINE VSIZ
50C     INCLUDE HYPE1
60C     INCLUDE HYPE2
100      PARAMETER A1=20
110      DIMENSION MCH(A1),MVT(A1),MPT(A1)
120      I50B=2
130      MCH(1)=MCH0
140      MPT(1)=MFB*MCH0
150      IF(MV.GT.0.)GO TO 95
160      V2=1.-KP*(PFE/ESP*KTP/TSP*PFM/MSP*PFW*KOL/CSP*
170     &        MFB/AE*PFFW*TFWP/(C*FWSF))
180      DO 50 I=2,A1
190      MVT(I-1)=(MCH(I-1)*MPL)/V2
200      MPT(I)=MVT(I-1)-MCH(I-1)-MPL
210      MCH(I)=KMP*(MPT(I)-MPT(I-1))*MCH(I-1)
220      MV=(MCH(I)*MPL)/V2
230      MVCH=MCH(I)
240      IF(ABS((MV-MVT(I-1))/MV).LT..001)GO TO 95
250C     WRITE(6,49)MV,V2,MVT(I),MCH(I),MPT(I)
260C 49  FORMAT(1X,5E10.3)
270  50  CONTINUE
280      WRITE(6,80)MV,MVT(A1-1),MCH(A1),MPT(A1),V2
290  80  FORMAT(1H,'VEH AT ROUTINE DNC',5E10.3)
300      STOP
310  95  PEMX=KP*MV*PFE
320      PMMX=KP*MV*PFM
330      TMMX=PMMX/WFMX*9549.3
331        PBASE=PMMX*500./EFMOT
332        IBASE=PBASE/VBASE
340      PFWMX=KP*MV*PFFW
350      ESBMX=MFB*MV*BSEC(BTYP)
360      PSBMX=MFB*MV*BSPC(BTYP)
370      MB=MFB*MV
380      RETURN
390      END
```

```
10C
20C     ********** DRIVING CYCLE ROUTINE **********
31C
40      SUBROUTINE DCYCLE
50C     INCLUDE HYPR1
60C     INCLUDE HYPR2
100      PARAMETER DC1=6,DC2=32,DC3=30,DC4=26
110      DIMENSION TDC1(DC1),TDC2(DC2),VDC1(DC1),VDC2(DC2)
120      DIMENSION TDC3(DC3),VDC3(DC3),TDC4(DC4),VDC4(DC4)
130        DATA TDC1/6.,0.,19.,38.,47.,72./
140       DATA VDC1/6.,0.,32.,32.,0.,0./
150       DATA TDC2/32.,0.,2.,4.,6.,8.,10.,12.,14.,16.,18.,20.,22.,24.,
160     &26.,28.,38.,58.,78.,80.,82.,84.,86.,88.,89.,90.,92.,94.,96.,
162     &97.,100.,123./
170       DATA VDC2/32.,0.,7.5,14.,18.8,22.8,26.2,29.2,29.2,31.9,34.3
180     &,36.5,38.4,40.3,42.,43.5,45.,44.96,44.94,44.9,42.2,39.4,36.5,33.5
190     &,30.3,26.9,23.6,16.9,10.3,3.8,0.,0./
220      DATA TDC3/30.,0.,25.,50.,67.,134.,142.,167.,180.,218.,233.,
230     &       240.,278.,295.,310.,325.,340.,470.,480.,555.,585.,615.,
240     &      630.,642.,655.,690.,715.,740.,762.,770./
250      DATA VDC3/30.,0.,36.,38.,48.,48.,39.,48.,43.,43.,47.,48.,44.,
260     &       28.5,44.,48.,58.,58.,55.,55.,48.5,48.5,55.,46.,51.,
270     &       52.,59.,47.,0.,0./
280      DATA TDC4/26.,0.,8.5,26.,29.,36.,44.5,63.,66.,74.,82.5,100.,
290     &       103.,110.,118.5,137.,140.,148.,153.,168.,
300     &      172.,178.,193.,216.,220.,230./
310      DATA VDC4/26.,0.,14.8,10.,0.,0.,14.8,10.,2*0.,14.8,10.,2*0.,
320     &      14.8,10.,2*0.,9.8,6.,2*0.,19.8,14.2,2*0./
330       ISUB=3
331          IGO=1
332         TXT=T
340  50   GO TO (100,200,300,400,500,600,700,800),DCTYP
350  300  TFIN=TDC1(DC1)
360       CALL LINT(TDC1,VDC1,TXT,V,A)
370       GO TO 1000
380  400  TFIN=TDC2(DC2)
390       CALL LINT(TDC2,VDC2,TXT,V,A)
391         A=A*1.609
392         V=V*1.609
400       GO TO 1000
410  700    CALL EPACYC(TXT,V,A,TFIN,DCTYP)
420       A=A*1.60934
430       V=V*1.60934
440       GO TO 1000
450  800   CALL EPACYC(TXT,V,A,TFIN,DCTYP)
460       V=V*1.60934
470       A=A*1.60934
480       GO TO 1000
490  500  CALL EPACYC(TXT,V,A,TFIN,DCTYP)
500       A=A*1.60934
510       V=V*1.60934
520       GO TO 1000
530  600  CALL EPACYC(TXT,V,A,TFIN,DCTYP)
540       A=A*1.60934
550       V=V*1.60934
560       GO TO 1000
570  100  TFIN=TSS(1)
580       A=0.
590       V=SSVEL(1)
600       V0=-100.
```

```
      GO TO 1000
  200 TFIN=TSS(2)
      A=0.
      VO=-100.
      V=SSVEL(2)
C     DTP=CURRENT DT
 1000   IF(IGO.EQ.2) GO TO 2000
          VSAVE=V
          ASAVE=A
          IGO=IGO+1
        TXT=T+DTP
       KM=KM+V*DTP/3600.
        GO TO 50
 2000    VNEXT=V
         ANEXT=A
        V=VSAVE
        A=ASAVE
      RETURN
      END
```

```
10C
20C     ********** DRIVE SHAFT POWER ****
30C
40      SUBROUTINE DSPWR
50C     INCLUDE HYPR1
60C     INCLUDE HYPR2
100      ISUB=4
110C     MAX VEH VEL
120      CALL LIN3(-1.,10.,V,1.)
130C     MAX VEH ACCEL
140      CALL LIN3(-1.,11.,A,1.)
150      V1=CROL*MV*(1.+CROL1*.911344*V+CROL2*(.911344*V)**2)*V/36/.0978
160      V2=CD*AF*(V+VWIND)**2*V/77760.*(530./(460.+ATEMP))
170      V3=5*MV*V/36709.78
180      V4=MV*A*V/12960.
190      V5=DTP/3600.
200      K1=K1+V1*V5
210      K2=K2+V2*V5
220      K3=K3+V3*V5
230      K4=K4+ABS(V4*V5)
240      PWHL=V1+V2+V3+V4
250C     INT OF POS WHEEL NRG
260      EVD=EVD+AMAX1(PWHL*V5,0.)
270C     INT OF NEG WHEEL NRG
280      PFMA=PFMA+AMIN1(PWHL*V5,0.)
290      WDS=V*5.30516/DWHL*GRATS(5)
300C     WHEEL INERTIAL POWER
310  200 PFEA=JWHL*V*A/(3240.*DWHL**2)
320      V1=PWHL+PFEA
330      IF(V1).40,
340      PDS=V1/ND**SIGN(1.,V1)
350      IF(PDS)42,,
360C     MAX POS PDS
370      CALL LIN3(-1.8,PDS,1.)
380      RETURN
390C     MAX NEG PDS
400   42 CALL LIN3(-1.9,-PDS,1.)
410      RETURN
420   40 PDS=0.
430      RETURN
440      END
```

```
10C
20C     ********** TRANSMISSION ROUTINE **********
30C
40      SUBROUTINE TRANS(IDUM)
50C     INCLUDE HYPR1
60C     INCLUDE HYPR2
100C     **********
110C     IDUM=1 NORMAL TRANSMISSION SHIFT F(VEHICLE VEL)
120C     IDUM=2 UP-SHIFT ONLY
130C     IDUM=3 INSUFFICIENT POWER DOWN-SHIFT
140C     **********
150      ISUB=5
160   222  CONTINUE
170         IF(ICVT.EQ.1) GO TO 900
180         IOGEAR=IGEAR
190C**    MODIFY OLD MODEL FOR CTTYP=2 TO INCLUDE GRM **
200C      WEMX=MAXIMUM ENGINE SPEED
210C      WMMX=MAXIMUM MOTOR SPEED
220   111   PRODT=SPER
230         IF(MON.EQ.0) PRODT=.333
240   444   VTMPE=PRODT*WEMX*DWHL/(GRATS(5)*5.3052)
250         VTMPM=VTMPE/GRM
260         VMAX=VTMPE
270         IF(VTMPM.LT.VTMPE)VMAX=VTMPM
280C
290 333     DO 500 I=1,3
300         SHIFT(I)=VTMPE/GRATS(I)
310        IF(DTTYP.EQ.1)GOTO 500
320         SHIFT(I)=VMAX/GRATS(I)
330    500 CONTINUE
340        IF(ITER.GT.1) GO TO 33
350         DO 550 I=1,3
360 550    SHIFTA(I)=SHIFT(I)*SPERA/PRODT
370    33 GO TO (40,10,400),IDUM
380    40 GO TO (100,200,300),TTYP
390  100 IF(PDS.GE.0.)GO TO 10
400        IF(IDOWN.LT.1)RETURN
410        IF(V.LE.SHIFT(1)-2.)IGEAR=1
420        IF((V.GT.SHIFT(1)-2.).AND.(V.LE.SHIFT(2)-2.))IGEAR=2
430        IF((V.GT.SHIFT(2)-2.).AND.(V.LE.SHIFT(3)-2.))IGEAR=3
440        IF(V.GT.SHIFT(3)-2.)IGEAR=4
450        GO TO 35
460    10 DO 20 I=1,3
470        IGEAR=I
480    20 IF(V.LE.SHIFT(I))GO TO 35
490        IGEAR=4
500  35      IF(IGEAR.EQ.1.OR.DTTYP.EQ.1) GO TO 45
510          IMUST=IGEAR-IOGEAR
520          IF(JFLUX.EQ.1.AND.IMUST.EQ.0)  IGEAR=IGEAR-1
530           JFLUX=0
540    45  NT=NTM(IGEAR)
550        GRT=GRATS(IGEAR)
560        RETURN
570  200  CALL ERR2
580  300  CALL ERR2
590  400  IF(IGEAR.EQ.1)RETURN
600        V1=WOS*GRATS(IGEAR-1)
610        GO TO (123,123,323),DTTYP
620  123  IF(V1.GT.WEMX)RETURN
630        IF(V1.GT.WMMX.AND.DTTYP.EQ.2)RETURN
```

```
640       OAR(1,192)=(1,192)+1.
650       WT=V1
660       IGEAR=IGEAR-1
670       WE=V1
680       EON=1
690       CALL ENG(1)
700       IF(DTTYP-1),,133
710       RETURN
720  133  MON=1
730       WM=WDS*GRATS(IGEAR)*GRM
740       CALL MTR(1)
750       RETURN
760  323  CALL ERR2
770       RETURN
780C
790C          ICVT=1 FOR CVT(CONTINOUSLY VARIABLE TRANSMISSION)  OPERATION
800C
810   900    IDO=1
811          IF(WDS.EQ.0.0) WDS=0.00000001
820         IF(ITER.EQ.0) GRT=RR
830         WEX=WDS*GRT
840          WMX=WDS*GRT*GRM
850        IF(WMX.LE.0.0) WMX=WIDLEM
860         IF(WEX.LE.0.0) WEX=WIDLE(ETYP)
861         NT=ECVT
870        IF(CTYP.EQ.3) GO TO 930
880         IF(CTYP.EQ.4) GO TO 940
890         GR1=RR
900         GR2=RR
910   910   IF(WEX.GT.WEMX) GR1=WEMX/WDS
920          IF(WMX.GT.WMMX) GR2=WMMX*SPFR/WDS
930         IF(WEX.GT.WEMX.OR.WMMX.LT.WMX) GRT=AMIN1(GR1,GR2)
940         GO TO 950
941  930    IF(WE.GT.WEMX) GRT=WEMX/WDS*SPFR
942          GO TO 950
950   940   IF(WMX.GT.WMMX) GRT=WMMX*SPFR/WDS
960   950   RETURN
970        END
```

```
10C
20C     ********** COMPONANT POWER ROUTINE **********
30C
40      SUBROUTINE COMPWR
50C     INCLUDE HYPR1
60C     INCLUDE HYPR2
100       ISUB=6
110       WT=WDS*GRT
120       IF(PDS).60.
130       PT=PDS/NT**SIGN(1.,PDS)
140       PJE=JENG*A*V*(GRT*GRATS(5))**2/(3240.*DWHL**2)
150       PJM=JMTR*A*V*(GRT*GRATS(5))**2/(3240.*DWHL**2)
160    65 GO TO (1000,2000,3000),DTTYP
170    60 PT=0.
180       GO TO 65
190  1000 WM=WDS*GRM
200       PJM=PJM*(GRM/GRT)**2
210  1005 WE=WDS*GRT
220       EON =1
230       MON=1
240       IMGF=0
250       IEIF=0
260          IF(DCTYP.GT.4) GO TO 1010
270       IF(VN.EQ.VU)GO TO 5000
280  1010 CALL ENG(1)
290       CALL MTR(1)
300       GO TO 5000
310  2000 WM=WDS*GRT*GRM
320       GO TO 1005
330  3000 CALL ERR2
340  5000 EON=0
350       MON=0
360       RETURN
370       END
```

```
C
C     ******** ENGINE SIMULATION SUBROUTINE ********
C
      SUBROUTINE ENG(ID2)
C     INCLUDE HYPR1
C     INCLUDE HYPR2
      PARAMETER SP1=7,PW1=8,NEM=10,
     &      SP2=9,PW2=8,
     &            SP3=10,PW3=11,SP4=10,PW4=11,
     &     SP5=11,PW5=11,SP6=9,PW6=11,
     &     SP7=10,PW7=11,SP8=9,PW8=11
     &            SP2=8,PW2=5,
C       ADD TO COMMON/RBLK2/EMHC,EMCO,EMNOX,NEHC,NECO,NENO,PFMX
      PARAMETER SP9=12,PW9=8,SP10=8,PW10=8
      DIMENSION SPEED9(SP9),POWER9(PW9),BSFC9(SP9,PW9),BSHC9(SP9,PW9),
     &BSCO9(SP9,PW9),BSNOX9(SP9,PW9),PMX9(SP9),PMX10(SP10),BSFC10(SP10,PW10)
C
      DIMENSION SPEED1(SP1),SPEED2(SP2),SPEED3(SP3),
     &       POWER1(PW1),POWER2(PW2),POWER3(PW3),
     &     BSFC1(SP1,PW1),BSFC2(SP2,PW2),BSFC3(SP3,PW3),
     &    PMX1(SP1),PMX2(SP2),PMX3(SP3),
     &     POWER4(PW4),SPEED4(SP4),PMX4(SP4),BSFC4(SP4,PW4),
     &   POWER5(PW5),SPEED5(SP5),PMX5(SP5),FUEL5(SP5,PW5),
     &   POWER6(PW6),SPEED6(SP6),PMX6(SP6),FUEL6(SP6,PW6),
     &   POWER7(PW7),SPEED7(SP7),PMX7(SP7),FUEL7(SP7,PW7),
     &   POWER8(PW8),SPEED8(SP8),PMX8(SP8),FUEL8(SP8,PW8),
     &POWER10(PW10),SPEED10(SP10),BSHC10(SP10,PW10),BSCO10(SP10,PW10)
     &  ,BSNOX10(SP10,PW10),BSCS10(SP10,PW10)
      DIMENSION HPMX(NEM)
      DIMENSION REVMX(NEM)
C
C     ********** ETYP=1 CHEVY VEGA 4-CYL  1975 **********
      DATA PMX1/0.,24.,36.,50.,70.,86.,100./
      DATA POWER1/0.,5.,10.,15.,25.,50.,75.,100.1/
      DATA SPEED1/0.,22.7,34.1,45.5,63.6,81.8,100.1/
      DATA BSFC1/3.,3.2,4.,2*3.2,2*3.1,2*2.8,3.6,2.81,2.85,2.68,
     &   2.7,2.2,1.95,1.84,1.52,1.45,1.49,1.64,1.45,1.32,1.45,
     &    1.11,1.03,1.12,1.17,1.,.979,1.04,.776,.73,.64,.873,
     &    .7,.596,.631,.526,.525,.617,.606,.5,.461,.5,.488,.497,
     &    .523,.553,.6,.583,.631,.576,.528,.542,.517/
C
C     ******* ETYP=2 DATA BELOW FROM BRIGGS & S 253417 *******
C     DATA POWER2/0.,25.,50.,75.,100.1/
C     DATA SPEED2/0.,50.,58.,3,66.7,75.,83.3,91.7,100.1/
C     DATA BSFC2/8*5.,2.,1.67,1.59,1.51,1.41,1.39,1.36,1.41,
C    &            1.5,1.07,1.09,1.04,.924,.897,.881,.905,
C    &         1.,.928,.89,.834,.752,.74,.737,.795,
C    &          1.,.852,.794,.74,.689,.67,.657,.654/
C     ****** DATA FOR CHEVY 350 CID 1975 ******
C
      DATA POWER2/0.,3.,10.,25.,40.,60.,75.,100.1/
      DATA SPEED2/5.,26.3,34.2,39.4,50.,57.9,73.7,86.8,100.1/
      DATA PMX2/3.,28.4,39.,48.,64.4,74.1,88.1,96.,100./
      DATA BSFC2/2*3.41,5.31,3.68,3.92,3.56,5.08,4.07,5.55,
     &    2*3.41,5.31,3.68,3.92,3.56,5.08,4.07,5.55,
     &     2*.954,1.48,1.31,1.36,1.33,1.57,1.54,1.76,
     &     2*.833,.698,.791,.715,.716,.822,.883,.804,
     &     2*.569,.542,.512,.552,.53,.525,.694,.692,
     &   2*.542,.477,.526,.5,.538,.527,.586,.629,
     &     2*.523,.463,.544,.511,.493,.532,.545,.618,
```

```
640     &     2*.548,.538,.442,.506,.497,.511,.536,.561/
650      DATA PMX3/0,.23.5,51.1,64.6,70.8,85.1,86.3,100.,92.6,89.8/
660C
670C     ********** ETYP = 3 DATA FROM HONDA CVCC 1975 **********
680      DATA POWER3/0,.10.,20.,30.,40.,50.,60.,70.,80.,90.,100.1/
690      DATA SPEED3/0,.18.2,36.4,45.,45.,54.5,63.6,72.7,81.8,90.9,100.1/
700      DATA BSFC3/10*5.,2.,1.7,1.9,1.5,1.9,4*2.,1.96,
710     &           1.,.98,.95,.855,.975,1.03,1.12,1.14,1.4,1.7,
720     &           .9,.86,.79,.697,.795,.785,.865,.81,.975,1.47,
730     &          .8,.76,.71,.61,.687,.559,.72,.655,.825,1.27,
740     &           .7,.655,.655,.554,.62,.581,.62,.585,.725,1.09,
750     &          .6,.572,.547,.508,.56,.54,.55,.554,.665,.942,
760     &         .6,.525,.491,.468,.51,.516,.516,.535,.625,.821,
770     &          .5,.497,.441,.44,.47,.505,.505,.525,.625,.729,
780     &          .5,.489,.443,.44,.472,.5,.506,.525,.625,.664,
790     &          .5,.483,.495,.49,.55,.5,.55,.525,.625,.627/
800C     DATA PMX2/0,.45.9,57.3,67.8,79.2,86.8,94.1,100./
810C     DATA REVMX/5000,.5500,.5500,.5600 ,4800,.4000,.4440,.3800./
820      DATA HPMX/90,.120,.51.1,98,.87.7,68.7,90,.105.6,72,.70./
830C
840C     **********ETYP=4  2.3L PINTO  1975*************
850C
860      DATA PMX4/0,.15.9,19.9,25.9,33.7,47.4,57.1,71.4,82.1,94.3/
870      DATA POWER4/0,.10.,20.,30.,40.,50.,60.,70.,80.,90.,100.0/
880      DATA SPEED4/0,.14.3,17.9,21.4,26.8,35.7,44.6,53.6,62.5,100.0/
890      DATA BSFC4/2*1.35,1.8,1.38,1.7,2.1,2.7,3.4,3.6,4.0,
900     &     2*1.12,1.25,1.07,1.,1.05,1.1,1.1,1.1,1.2,1.4,
910     &     2*.875,.89,.79,.68,.67,.69,.70,.73,.76,
920     &     2*.675,.67,.607,.56,.56,.58,.575,.59,.6,
930     &     2*.57,.56,.552,.51,.5,.51,.515,.53,.55,
940     &     2*.52,.5,.5,.465,.475,.475,.476,.483,.495,
950     &     2*.515,.47,.472,.45,.45,.45,.45,.47,.48,
960     &     2*.52,.46,.46,.43,.43,.433,.445,.45,.47,
970     &     2*.528,.47,.466,.425,.43,.432,.45,.45,.46,
980     &     2*.54,.495,.47,.42,.44,.431,.46,.465,.475,
990     &     2*.56,.475,.42,.45,.431,.465,.468,.48,.5/
1000C
1010C    **********ETYP=5  PINTO 140 CID  1977******
1020C
1030     DATA PMX5/0,.6,.6.96,17.4,30.9,44.1,56.7,66.6,75.6,86.8,100.01/
1040     DATA SPEED5/0,.15.62,17.71,20.83,31.25,41.67,54.17,62.5,
1050    &  72.92,83.33,100.1/
1060     DATA POWER5/0,.10.,20.,30.,40.,50.,60.,70.,80.,90.,100.1/
1070     DATA FUEL5/2.5,2.5,2.7,3.1,4.3,4.8,5.2,6.0,8.0,10.,14.2,
1080    &     2.5,2.5,2.8,3.2,4.5,4.795,6.4,8.157,9.243,11.563,16.179,
1090    &     2.6,2.6,2.9,3.3,4.8,5.511,7.7,9.824,10.699,14.044,18.701,
1100    &     2.65,2.65,3.,3.5,5.2,6.676,9.,12.066,13.349,16.,20.853,
1110    &     2.8,2.8,3.1,3.8,5.7,7.483,10.5,13.591,15.244,16.971,23.22,
1120    &     2.9,2.9,3.3,4.2,6.4,8.915,12.,15.369,17.506,19.046,25.809,
1130    &     3.,3.,3.5,4.7,7.2,10.611,13.7,16.819,19.496,22.33,28.98,
1140    &     3.2,3.2,3.8,5.3,8.3,11.592,15.7,17.206,19.804,25.404,33.501,
1150    &     3.4,3.4,4.2,6.4,9.9,13.297,18.3,19.469,23.027,28.468,40.39,
1160    &     3.6,3.6,4.7,8.4,12.5,16.77,21.9,25.489,30.903,34.035,47.988,
1170    &     4.,4.,5.3,12.4,16.5,21.,28.1,33.5,37.6,43.7,49.7/
1180C
1190C**********ETYP=6 MERCEDES DIESEL 183 CID  1975*****
1200C
1210     DATA PMX6/0,.20.,31.3,44.1,60.3,75.1,91.4,94.6,100./
1220     DATA SPEED6/0,.17.5,27.5,37.5,50.,60.,75.,87.5,100.1/
1230     DATA POWER6/0,.10.,20.,30.,40.,50.,60.,70.,80.,90.,100.1/
```

```
      DATA FUEL6/1.0,1.,1.4,2.9,3.4,5.5,8.5,9.,13.9,
     &    1.,1.,2.2,3.868,4.7,6.973,10.,11.,14.826,
     &    1.,1.,2.,5.4.932,6.,9.141,11.3,10.093,16.21,
     &    1.,1.,3.5,5.614,7.25,10.759,12.6,13.891,17.792,
     &    1.,1.,4.1,6.28,8.5,11.806,14.,17.204,18.997,
     &    1.,1.,4.5,7.181,9.5,12.604,15.5,19.188,19.817,
     &    1.05,1.05,5.5,8.33,11.1,13.948,17.2,20.967,22.066,
     &    1.1,1.1,6.3,9.6,12.5,16.311,19.1,23.416,26.758,
     &    1.2,1.2,7.1,10.576,13.9,18.313,21.7,25.005,30.322,
     &    1.3,1.3,8.1,11.649,15.4,19.186,26.8,27.06,32.432,
     &    1.4,1.4,9.4,13.4,18.8,19.5,27.6,32.,35.6/
C
C********ETYP=7 PONTIAC 151  1978*****
C
      DATA PWX7/0.,2.,22.5,35.,52.,64.,74.,88.,82.3,95.8,100./
      DATA SPEED7/0.,14.,22.,32.,34.09,45.,45.,56.81,63.63,72.,72.,
     &    86.,36.,100./
      DATA POWER7/0.,10.,20.,30.,40.,50.,60.,70.,80.,80.,100./
      DATA FUEL7/1.3,1.3,2.3,3.1,4.,5.,6.,7.,7.,9.4,10.8,
     &    1.391,1.391,2.81,4.,5.9,7.,8.5,9.,10.886,13.397,
     &  1.481,1.481,3.12,5.,7.2,8.686,10.4,11.985,13.506,15.97,
     &    1.578,1.578,3.736,5.9,8.5,10.587,12.2,14.037,16.577,18.759,
     &    1.682,1.682,4.608,6.9,9.8,12.432,14.,16.11,19.308,21.3,
     &    1.783,1.783,5.249,7.9,11.3,14.163,15.9,18.174,22.122,23.95,
     &    1.877,1.877,5.696,9.1,12.7,16.1,17.9,20.883,25.492,27.6,
     &    1.957,1.957,6.162,10.3,14.4,18.388,20.2,24.404,29.492,32.382,
     &    2.015,2.015,6.947,11.7,16.3,20.232,22.6,26.435,32.699,35.218,
     &    2.056,2.056,8.374,13.5,18.5,22.9,26.5,29.117,36.384,38.8,
     &    2.1,2.1,10.4,17.1,22.5,17.9,32.2,36.2,42.6,47.5/
C
C*****ETYP=F CHEVY 250 6 CYL  1975 *******
C
      DATA PWX8/0.,5.4,12.6,44.8,56.67,77.18,87.03,96.87,100./
      DATA SPEED8/0.,14.5,23.7,31.5,39.5,55.3,71.,99.,100./
      DATA POWER8/0.,10.,20.,30.,40.,50.,60.,70.,80.,90.,100./
      DATA FUEL8/3.4,3.4,3.6,4.,5.3,8.6,10.5,14.,16.,
     &    3.4,3.4,4.112,5.533,6.8,11.7,12.5,19.201,17.958,
     &    3.4,3.4,3.913,7.115,8.533,14.637,14.987,21.06,21.51,
     &    3.4,3.4,3.996,8.341,10.418,17.196,18.05,22.321,23.825,
     &    3.4,3.4,4.032,9.043,12.625,19.5,21.41,27.751,26.484,
     &    3.5,3.5,4.456,10.242,15.327,21.779,23.968,31.217,32.081,
     &    3.6,3.6,5.071,11.833,17.923,23.3,26.5,33.776,36.234,
     &    3.6,3.6,5.577,13.462,19.659,23.307,29.897,37.759,38.675,
     &    3.8,3.8,5.707,14.891,20.082,24.265,34.023,41.921,48.228,
     &    4.,4.,5.888,17.876,22.046,29.134,39.966,46.646,58.274,
     &    4.5,4.5,6.7,23.3,27.9,38.1,48.,54.,64.5/
C
C        VW ENGINE ETYP=9***************
         DATA POWER9/10.,14.,29.,43.,57.,72.,86.,100./
         DATA SPEED9/15.,20.,26.6,33.3,40.,47.,53.,60.,67.,76.7,
     &    86.7,100./
         DATA PWX9/11.1,19.4,26.3,38.2,45.8,53.6,61.1,69.,76.8,
     &    88.1,100.,100./
         DATA BSFC9/3.06,1.98,1.76,1.39,1.29,1.2,1.,1.29,1.33,1.29,1.36,1.
     &    67.2,15.1.6,1.5,1.12,1.119,1.1,1.1,1.1,1.194,1.219,1.179,1.242,1.5,
     &    61.4,1.051,.866,.743,.727,.715,.72,.75,.742,.765,.789,.9,
     &    61.03,.759,.66,.579,.595,.582,.588,.597,.605,.608,.63,.746,
     &    6.80,.617,.539,.521,.513,.513,.533,.521,.525,.55,.566,.642,
     &    6.72,.566,.513,.495,.496,.495,.493,.479,.492,.507,.526,.59,
     &    6.65,.526,.486,.474,.461,.463,.472,.467,.467,.485,.505,.554,
```

```
1840      &.65,.521,.476,.446,.447,.445,.452,.447,.452,.468,.487,.528/
1850       DATA BSHC9/8,.3,1,3,4,3,55,3,6,3,6,3,65,3,5,3,4,3,1,3,0,3,02,
1860      &6,.3,.3,25,3,33,3,4,3,4,3,48,3,3,3,24,2,93,2,79,2,92,
1870      &4,3,2,42,2,66,2,65,2,73,2,68,2,81,2,62,2,52,2,42,2,24,2,58,
1880      &3,7,2,13,2,32,2,31,2,56,2,49,2,47,2,19,2,34,2,08,2,02,2,3,
1890      &3,7,1,97,2,03,2,26,2,37,2,18,2,27,2,11,2,81,1,99,1,91,2,05,
1900      &4,68,1,85,1,79,1,98,2,05,1,98,2,13,1,89,1,85,1,77,1,67,1,78,
1910      &4,29,1,82,1,7,1,94,1,93,1,75,1,95,1,72,1,66,1,62,1,53,1,54,
1920      &5,.2,1,1,75,1,81,1,94,1,99,1,77,1,43,1,52,1,48,1,37,1,38/
1930       DATA BSC09/120,.60,.47,.35,.43,.40,.46,.55,.54,.51,.45,.74,.
1940      &85,.49,.40,5,32,8,37,.36,4,41,.46,9,47,7,45,5,38,7,67,.
1950      &43,.27,.27,.23,4,24,.26,7,29,6,32,5,28,8,27,.26,4,46,7,
1960      &37,.19,.19,.17,7,22,7,23,1,24,5,23,5,23,2,20,3,20,2,30,8,
1970      &35,.16,6,17,6,18,.20,7,20,4,20,3,19,3,17,9,18,2,18,2,21,6,
1980      &32,.15,5,17,1,18,9,20,7,18,.17,5,17,4,16,6,17,2,17,.19,.
1990      &87,.17,3,16,9,19,2,17,3,16,2,15,8,15,8,16,2,15,7,15,7,17,6,
2000      &150,.29,8,17,6,16,8,16,7,15,3,14,9,15,9,15,4,14,4,14,7,15,9/
2010       DATA BSNOX9/2,2,2,55,2,75,2,8,4,5,6,6,9,.9,2,13,4,17,8,21,8,16,.
2020      &2,3,2,6,2,9,4,2,6,2,7,1,9,5,9,83,13,9,19,3,22,4,17,5,
2030      &2,5,2,8,3,75,6,4,8,5,9,3,11,7,12,8,15,7,21,7,23,8,22,3,
2040      &2,7,3,3,5,1,8,9,11,1,11,6,14,2,15,7,18,3,22,4,24,8,25,.
2050      &3,2,4,2,6,35,11,4,13,4,14,1,16,4,17,.19,3,22,8,25,3,24,8,
2060      &5,5,6,75,7,25,11,1,13,2,13,5,15,5,16,.18,.20,3,22,5,23,2,
2070      &6,5,7,35,8,2,12,1,14,1,14,5,16,2,16,9,17,8,21,1,22,6,23,.
2080      &8,.8,64,9,6,12,9,14,9,15,5,17,3,17,5,18,7,22,1,23,0,23,5/
2090C
2100       DATA PMX10/16,8,28,8,47,6,63,0,76,6,94,6,98,.100,/
2110       DATA SPEED10/20,.30,.40,.50,.60,.80,.90,.100,1/
2120       DATA POWER10/10,.20,.30,.40,.50,.60,.80,.100,1/
2130       DATA BSFC10/1,08,.97,.84,.84,.84,.95,1,075,1,15,.85,.74,
2140      &,66,.66,.66,.76,.885,1,0,
2150      &,69,.60,.54,.54,.54,.64,.74,.835,
2160      &,58,.515,.485,.485,.485,.565,.64,.725,
2170      &,540,.470,.465,.465,.465,.51,.583,.65,
2180      &,515,.465,.45,.45,.45,.485,.54,.605,
2190      &,50,.455,.445,.445,.445,.465,.495,.565,
2200      &,525,.49,.48,.48,.48,.49,.50,.57
2210      &/
2220       DATA BSHC10/,50,.50,.50,.53,.53,.45,.23,.26,.38,
2230      & ,38,.38,.41,.45,.62,.50,.29,.30,.30,.30,.34,.41,.87,.60,.315,
2240      & ,24,.22,.22,.26,.37,1,08,.65,.34,.21,.195,.175,.19,.30,1,08,
2250      & ,69,.35,.20,.175,.145,.14,.22,.86,.71,.37,.17,.15,.10,.10,.13,.45,
2260      & ,61,.35,.14,.10,.08,.08,.08,.20,.29,.26/
2270       DATA BSNOX10/5,7,5,7,5,7,6,2,7,3,10,.10,3,10,4,4,4,4,4,4,4,
2280      & 5,2,6,2,8,8,9,1,9,0,3,7,3,7,3,7,4,2,4,7,8,1,8,4,7,7,
2290      & 3,0,3,0,3,0,3,4,3,9,6,6,7,3,6,8,2,6,2,6,2,6,3,0,3,4,5,9,6,8,6,2,
2300      & 2,4,2,4,2,4,2,7,3,1,5,3,6,2,5,6,1,8,2,0,2,0,2,3,2,7,4,0,4,8,4,5,
2310      & 1,2,1,5,1,5,1,9,2,2,2,8,3,1,3,1/
2320       DATA BSC010/4,0,3,7,3,1,3,5,4,5,.5,.25,.55,3,2,2,8,2,2,2,6,
2330      & 3,4,.55,.30,.55,2,6,2,1,1,6,1,9,2,4,.80,.45,.55,2,3,1,6,1,2,
2340      & 1,4,2,0,1,0,.60,.55,2,3,1,5,1,1,1,2,1,9,1,0,.70,.50,
2350      & 2,8,1,5,1,1,1,1,1,8,1,0,.75,.37,4,4,2,1,1,1,1,1,1,.5,
2360      & ,60,.37,6,7,6,7,1,9,1,5,1,3,.25,.30,.26/
2370       DATA BSCS10/14,0,6,0,6,0,5,3,6,0,10,5,10,5,12,0,12,0,4,1,4,1,
2380      & 3,6,4,3,9,3,9,3,10,5,10,8,2,9,2,9,2,6,3,4,8,1,8,1,8,1,
2390      & 10,8,2,5,2,5,2,3,3,0,7,2,7,2,6,0,12,0,2,6,2,4,2,1,2,9,6,3,
2400      & 6,3,4,3,13,4,3,5,2,5,1,9,2,8,5,2,5,2,3,4,18,0,7,4,3,2,1,6,2,2,3,0,
2410      & 3,0,2,1,25,.10,0,8,5,2,3,2,7,3,2,2,3,1,5/
2420C     ********** ENGINE OPER MODE **********
2430      ISUB=9
```

```
      MM=ETYP
      IF(IEON)55,55,
      IF(WE.LT.WIDE)WE=WIDE
      WF=WE/WEMX*100.
         IF(WF.GT.100.)WF=99.999999
      IF(ID2.EQ.1)GO TO 1000
      IF(IESF.EQ.1)FUEL=FUEL+FSTRT(MM)*PEMX*608.28/HPMX(MM)
      IESF=0
      IF(IEIF),,50
      CALL LIN3(-1,1,PE,1.)
      CALL LIN3(-1,2,WF,1.)
      IF(PF)60,,
   54 GO TO (100,200,300,400,500,600,700,800,850,870),ETYP
   50 FCONS=FIDLE(MM)*PEMX*608.28/HPMX(MM)*DTP/3600.
      IEON=IEON+1
      FUEL=FUEL+FCONS
      WE=WIDE
      PE=0.
      EENG=0.
      RETURN
   55 WE=0.
      PE=0.
      HSFC=0.
      EENG=0.
      FCONS=0.
        IF(IHD.EQ.0) RETURN

        HEATER  &   DEFROSTER  POWER REQUIREMENTS
        QREQD=QHEAT*DTP/(3600.*QCONST)
        FCONS=QREQD*454.5
        FUEL=FUEL+FCONS
      RETURN
   60 WRITE(6,62)PE,PMXE,T,IGEAR,ICYC,DCTYP,I1
   62 FORMAT(1X,'ENG PWR NEG:  ',F7.2,2F7.1,4I5)
      PE=0.
      GO TO 54

C     ******** ENGINE # 1  BRANCH ********
  100 CALL PFCAL(PMXE,PEMX,PF)
      CALL BLIN(SPEED1,POWER1,BSFC1,SP1,PW1,WF,PF,BSFC)
      GO TO 900

C     ****** ENGINE #2 BRANCH *******
  200 CALL PFCAL(PMXE,PEMX,PF)
      CALL BLIN(SPEED2,POWER2,BSFC2,SP2,PW2,WF,PF,BSFC)
      GO TO 900

C     ********** ENGINE #3 BRANCH ********
  300 CALL PFCAL(PMXE,PEMX,PF)
      CALL BLIN(SPEED3,POWER3,BSFC3,SP3,PW3,WF,PF,BSFC)
      GO TO 900
C     *********ENGINE #4 BRANCH*********
  400 CALL PFCAL(PMXE,PEMX,PF)
      CALL BLIN(SPEED4,POWER4,BSFC4,SP4,PW4,WF,PF,BSFC)
      GO TO 900
  500 CALL PFCAL(PMXE,PEMX,PF)
      CALL BLIN(SPEED5,POWER5,FUEL5,SP5,PW5,WF,PF,BSFC)
      GO TO 920
  600 CALL PFCAL(PMXE,PEMX,PF)
      CALL BLIN(SPEED6,POWER6,FUEL6,SP6,PW6,WF,PF,BSFC)
```

```
2980      GO TO 920
2990  700 CALL PFCAL(PMXE,PEMX,PF)
3000      CALL BLIN(SPEED7,POWER7,FUEL7,SP7,PW7,WF,PF,BSFC)
3010      GO TO 920
3020  800 CALL PFCAL(PMXE,PEMX,PF)
3030      CALL BLIN(SPEED8,POWER8,FUEL8,SP8,PW8,WF,PF,BSFC)
3040      GO TO 920
3050C
3060C      *** EMISSION CALCULATIONS ***
3070C      NC=HYDROCARBONS, CO = CARBON MONOXIDE, NOX=NITRIC OXIDE
3080C      NEHC,NECO,NENO = CATALYST CONVERSION EFFICIENCY
3090C      EMHC,EMCO,EMNOX = GRAMS OF EMISSIONS
3100C
3110C      EMISSION CALCULATIONS
3120C      COME TO 850 IF ETYP=9
3130    850 CALL PFCAL(PMXE,PEMX,PF)
3140        CALL BLIN(SPEED9,POWER9,BSFC9,SP9,PW9,WF,PF,BSFC)
3150        CALL BLIN(SPEED9,POWER9,BSHC9,SP9,PW9,WF,PF,BSHC)
3160        CALL BLIN(SPEED9,POWER9,BSCO9,SP9,PW9,WF,PF,BSCO)
3170        CALL BLIN(SPEED9,POWER9,BSNOX9,SP9,PW9,WF,PF,BSNOX)
3180        DTXZ=DTP*PE*1.34102/3600.
3190        TCHW=TCHW0*(1.0-EXP(-TSAV/CTCT))
3200        EFAT=1.0
3210       EBOTA=ETON/TCHW
3220       IF(EBOTA.LT.20.) EFAT=1.0-EXP(-EBOTA)
3230        EF=EFOHC*EFAT
3240        EMHC=EMHC+BSHC*(1.0-EF)*DTXZ
3250        EF=EFOCO*EFAT
3260        EMCO=EMCO+BSCO*(1.0-EF)*DTXZ
3270        EF=EFONOX*EFAT
3280        EMNOX=EMNOX+BSNOX*(1.0-EF)*DTXZ
3290       GO TO 900
3300  870  CALL PFCAL(PMXE,PEMX,PF)
3310       CALL BLIN(SPEED10,POWER10,BSFC10,SP10,PW10,WF,PF,BSFC)
3320        CALL BLIN(SPEED10,POWER10,BSHC10,SP10,PW10,WF,PF,BSHC)
3330        CALL BLIN(SPEED10,POWER10,BSNOX10,SP10,PW10,WF,PF,BSNOX)
3340        CALL BLIN(SPEED10,POWER10,BSCO10,SP10,PW10,WF,PF,BSCO)
3350        CALL BLIN(SPEED10,POWER10,BSCS10,SP10,PW10,WF,PF,BSCS)
3360        DTXZ=DTP*PE*1.34102/3600.
3370       EMHC=EMHC+BSHC*DTXZ
3380        EMCO=EMCO+BSCO*DTXZ
3390        EMNOX=EMNOX+DTXZ*BSNOX
3400        EMCS=EMCS+BSCS*DTXZ
3410C
3420  900 FCONS=BSFC*PE*608.28*DTP/3600.
3430C
3440C       HEATER/DEFROSTER
3450        BURNM=0.0
3460       TIMM=3600.
3470        IF(IHD.EQ.0) GOT O 905
3480        EFFENG=0.134/BSFC
3490        ENGM=FCONS/454.5
3500       QOUT=(1.0-EFFENG)*ENGM/3.
3510        QREQD=QHEAT*DTP/(QCONST*TIMM)
3520        QNET=QREQD-QOUT
3530       IF(QNET.LT.0.0) GO TO 905
3540        BURNM=QNET*454.5
3550C
3560  905 FUEL=FUEL+FCONS+BURNM
3570      EENG=2545./(BSFC*190.)
```

```
3580  910  IF(DTP.EQ.DT)IEON=IEON+1
3590       OAR(5,193)=OAR(5,193)+PF/100.
3600      OAR(6,193)=OAR(6,193)+WF/100.
3610       TEENG=TEENG+PE*DTP/3600.
3620       CALL LIN3(-1,12,PF,1.)
3630       CALL LIN3(-1,13,WF,1.)
3640       RETURN
3650C      GRAMS
3660  920  FCONS=BSFC*DTP/3600.*453.592*1.34102*PEMX/HPMX(ETYP)
3670       FUEL=FUEL+FCONS
3680       IF(BSFC.GT..001)EENG=2545.*PE*1.34102/(BSFC*190.)
3690       GO TO 910
3700 1000  CALL ACC
3710       GO TO (1010,1020,1030,1040,1050,1060,1062,1064,1066,1068),ETYP
3720 1010  CALL LIN2(SPEED1,PMX1,SP1,WF,PMXE)
3730       GO TO 1070
3740 1020  CALL LIN2(SPEED2,PMX2,SP2,WF,PMXE)
3750       GO TO 1070
3760 1030  CALL LIN2(SPEED3,PMX3,SP3,WF,PMXE)
3770       GO TO 1070
3780 1040  CALL LIN2(SPEED4,PMX4,SP4,WF,PMXE)
3790       GO TO 1070
3800 1050  CALL LIN2(SPEED5,PMX5,SP5,WF,PMXE)
3810       GO TO 1070
3820 1060  CALL LIN2(SPEED6,PMX6,SP6,WF,PMXE)
3830       GO TO 1070
3840 1062  CALL LIN2(SPEED7,PMX7,SP7,WF,PMXE)
3850       GO TO 1070
3860 1064  CALL LIN2(SPEED8,PMX8,SP8,WF,PMXE)
3870       GO TO 1070
3880 1066   CALL LIN2(SPEED9,PMX9,SP9,WF,PMXE)
3890          GO TO 1070
3900 1068   CALL LIN2(SPEED10,PMX10,SP10,WF,PMXE)
3910 1070  PMXE=PMXE*PEMX/100.
3920       RETURN
3930       END
```

```
010 SUBROUTINE MTR(IDI)
020C
025C ********************************************************
027C THIS SUBROUTINE IS USED TO SIMULATE THE ELECTRIC DRIVE SYSTEM OF A
030C  HYBRID VEHICLE
100C
110       ISUB=7
120       IF(WM.LE.WIDM)WM=WIDM
130       WF=WM/WMMX*100.
140C
150C PROGRAM VARIABLES
160C IBASE = BASE CURRENT
170C IARM = ARMATURE CURRENT
180C IFLD = FIELD CURRENT
190C IBATT = BATTERY CURRENT
200C IAUX = AUXILLIARY CURRENT
210C IMGF = 0 IF MOTORING NO REGENERATION
220C IMFG = 1 IF BRAKING/REGENERATION
230C JELEC = 1 DC MOTOR WITH SEPERATE ARMATURE AND FIELD CHOPPER PCU
240C      AND FIELD CHOPPER PCU
250C JELEC = 3 DC MOTOR WITH ONLY ONE CHOPPER USED IN STARTING
260C      MOTOR BY CONNECTING TO ARMATURE AND LATER SWITCHING
270C      TO FIELD CIRCUIT
280C JELEC = -1 AC MOTOR - INDUCTION MOTOR DRIVE SYSTEM
290C JTORQ = 1 IF DESIRED POWER/TORQUE CAN BE PRODUCED
300C       = 0 IF DESIRED ELECTRIC POWER/TORQUE CANNOT BE PRODUCED
310C MODE = 1 IF ARMATURE VOLTAGE CONTROL IS BEING USED FOR DC MOTOR
320C MODE = 2 IF FIELD VOLTAGE CONTROL IS BEING USED FOR DC MOTOR
330C MODE = 3 FIELD CONTROL WITH FLUX SET AT MINIMUM VALUE
340C MON = 0 MOTOR IS OFF COMPLETELY
350C MON = 1 MOTOR IS ON PRODUCING POWER
360C MON = 2 MOTOR IS ON. FIELD EXCITED NO ARMATURE VOLTAGE.
370C           ROTOR SPINNING AT SPEED WIDLE
380C MON = 3 MOTOR FIELD EXCITED NO SPINNING, NO ARMATURE EXCITED
390C KE = SPEED CONSTANT
400C AKT = TORQUE CONSTANT
410C FLUX = FLUX
420C HM=INERTIA CONSTANT
430C RST = STARTING RESISTOR
440C RA = ARMATURE RESISTANCE
450C RB = BATTERY RESISTANCE
460C RF = FIELD RESISTANCE
470C PE = TOTAL ELECTRICAL POWER DEVELOPED
480C WFL  = FRICTION POWER
490C W = WINDAGE LOSS POWER
500C WCL = CORE LOSS PO
510C WSL = STRAY LOAD LOSS POWER
520C WEL = TOTAL POWER REQUIRED TO OVERCOME LOSSES
530C PMECH = TOTAL MECHANICAL TORQUE OR POWER REQUIRED
540C      PBASE=BASE POWER IN WATTS
550C T = TIME
560C DT = TIME STEP SIZE
570C WCHOP = CHOPPER POWER LOSS
580C CEMF = MOTOR BACK EMF
590C VARM = ARMATURE VOLTAGE
600C VCHOP = CHOPPER VOLTAGE
610C EBATT = BATTERY VOLTAGE
620C VBRUSH = VOLTAGE DROP ACROSS BUSHES
630C WM/RPM = MOTOR SPEED
640C ACCN = MOTOR ACCELERATION
```

```
650C IS MOTOR/GEN ON?  MON = 0 IF MOTOR OFF
651         RSS=10.
660         IF(MON.GT.1) GO TO 200
670         WM=0.0
680         PM = 0.0
690         FLUX=0.0
700         VCHOP=0.0
710         IARM=0.0
720         IBATT=0.0
730         PBATT=0.0
740         PBAT2=0.0
750         ABAT2= .0
760         IFLD = 0.0
770         GOTO 999
780 200     IF(ID1.EQ.1) GO TO 400
790C WHAT TYPE OF MOTOR?
800C IF(JELEC.LE.-1)CALL ACMOTO
810         IF(JELEC.GE.1)CALL DCMOTO
820         IF(DT.EQ.DTP)IMON=IMON+1
830         GO TO 999
840  400    PFAT=1.0
841         CALL ACC
850          IF(ITR.EQ.1) CALL DCMOTO
860         IF(IBTYP.EQ.2) PFAT=.5
870         IF(IBTYP.EQ.3) PFAT=0.9
880          PMAX=PMMX*PFAT
890         PMXM=PMAX
900         IF(WM.LT.WBASE)PMXM=PMAX*WM/WBASE
910         IF(JELEC.EQ.1.OR.IBS.EQ.0) GO TO 999
920        IF(WM.LT.WBASE) PMXM=0.4*PMAX
930          IF(JACCEL.EQ.1) CALL DCMAX(PMXM,WM,SNEW,RSS)
940 999      IF(ITR.EQ.1) CALL DCMOTO
950        RETURN
960         END
970         SUBROUTINE DCMAX(PMXM,WM,SNEW,RSS)
980         DATA RSTART ,EBO,RBB,FLUX/.076,113.4,.055,.9/
990         DATA PLOSS,CCL,CF,CW,RA,AKT,AKV,RL,RBD,VBO/346.6,156.,236.,28.8,.0235,
1000       &.3667,.052,.007,.00233,1.15/
1010          WB=2000.
1020 500      WCRIT=2000.
1021        WXX=WM
1022        IF(WXX.LT.1000.) WXX=1000.
1030         RTT=RA+RL+RBD
1040         PMLOSS=CF*WXX/WB+CCL*(WXX/WB)**1.5+CW*(WXX/WB)**3.
1050 550     IF(WXX.GT.2000.) GO TO 570
1060         EBB=EBO/2.-RSS*SNEW/2.
1070         RB=RBB/4.
1080         RST=0.0
1100         GO TO 600
1110 570     EBB=EBO-RSS*SNEW
1120         RB=RBB
1130         RST=0.0
1140        ALOSS=PLOSS/EBB
1180 600       FLUX=(EBB-RB*ALOSS-VBO)/(2.*AKV*WXX)
1181         IF(FLUX.GT.0.9) FLUX=0.9
1190 700     ALOSS=PLOSS/EBB
1200         ACUR=(EBB-VBO-RB*ALOSS-AKV*WXX*FLUX)/(RB+RTT+RST)
1210         PEM=AKT*WXX*FLUX*ACUR/7047.
1220         PMAX=(PEM-PMLOSS/1000.)/1.01
1230         PMXM=PMAX*.9
```

```
1231          TEM=AKT*ACUR*FLUX
1232          TMX=TEM-PMLOSS*7.047/WXX
1233          ABAT=ACUR*ALOSS
1234          ER=EBB-RH*ABAT
1235  888     II=II+1
1236          JJ=II/8
1237          KK=II-JJ*8
1238          TIM=II/H.
1240  999     RETURN
1250  END
```

```
010C ********************************:*********************
020        SUBROUTINE BAT(INBAT)
030C ************************************************************
040C
050C      THIS SUBROUTINE IS USED TO COMPUTE
060C          (1) THE BATTERY CURRENT + VOLTAGE FOR A GIVEN BATTERY POWER
070C          (2  THE PARAMETERS (E0,E1,E2) OF THE SIMPLIFIED BATTERY MODEL
080C              GIVEN BY EBATT = E0-31*SS-E2*IBATT
090C
091C      INCLUDE HYPR1
092C      INCLUDE HYPR2
100C
110        DIMENSION ECELL(3)
120         DATA AR1,AR2,AR3,BR1,BR2/6.556,0.1,-.714,.352,.127/
130C      IBATT = TOTAL BATTERY CURRENT
140C      ABATU = UNIT BATTERY CURRENT
150C      PBATT = TOTAL BATTERY POWER
160C      EBATT = TOTAL BATTERY VOLTAGE
170C
180C     IF HERE FOR THE FIRST TIME  INITIALIZE
190          IF(ITR.NE.1) GO TO 101
200C
210C      CONVERT BATTERY PARAMETERS
220C
230          RBASE=VBASE/IBASE
240         VNZ=VCNZ/VBASE
250         VNF=VCNF/VBASE
260         RNF=RCNF/RBASE
270         RNZ=RCNZ/RBASE
280          VBB=6.3*VBASE/EBREF
290          ALPHAT(1)=ALPHA(1)/VBASE
300          ALPHAT(2)=ALPHA(2)/VBASE
310          ALPHAT(3)=ALPHA(3)
320          ALPHAT(4)=ALPHA(4)
330          ALPHAT(5)=ALPHA(5)/VBASE
340          BETAT(1)=BETA(1)/RBASE
350          BETAT(2)=BETA(2)/RBASE
360          BETAT(3)=BETA(3)*IBASE
370          BETAT(4)=BETA(4)*IBASE
380          BETAT(5)=BETA(5)/RBASE
390          EBVE2=EBVEH/VBASE
400          EIT=EII/VBASE
410C     SET INITIAL VALUES
420            SS = 0.0
430            GO TO (10,20,30),IBTYP
440  10        E0 = ALPHAT(2)*NS
450            E1 = (ALPHAT(2)-EIT)*NS
460            E2 = -BETAT(2)*NS/NP
470           GO TO 100
480   20       E0=VNF*NC1*NS
490           E1=0.0
500           E2=RNF*NC1*NS/NP
510           GO TO 100
520    30      E0=VNZ*NC1*NS
530            E1=0.0
540            E2=RNZ*NC1*NS/NP
550  100       A1=AR1*EBREF/(VBASE*ALPHA(2))
560           A2=AR2*EBREF/(VBASE*ALPHA(2))
570           A3=AR3*EBREF/(VBASE*ALPHA(2))
580           B1=BR1*EBREF/(VBASE*ALPHA(2))
```

```
590       B2 =BR2*EBREF/(VBASE*ALPHA(2))
600       B3=BR3*EBREF/(VBASE*ALPHA(2))
610         ITBAT=0
620C       IF BATTERY VOLTAGE IS KNOWN SKIP THIS SECTION
630 101    IF(INBAT-1)300,200,500
640C
650C      SOLVE QUADRATIC EQUATION TO GET IBATT
660C
670    200 AA=E2
680        BB=E1*SS-E0
690        CC=PBAT2*NC1*NS*NP*UAHC/(NP2*NC2*NS2*UAHC2)
700       TMP=BB*BB-4.*AA*CC
710       IF(TMP.GE.0.0) GO TO 220
720       TMP=0.0
730       ITBAT=ITBAT+1
740        IF(ITBAT.GT.1) GO  TO 220
750        WRITE(6,229) ITER,PM,WM,PE,WE,FLUX,IARM,ABAT2,PBAT2,E0,E2,EBAT2
760      &,AA,BB,CC,NP2,NS2,PMXM
770 229   FORMAT('ITER       PM          WM          PE          WE     FLUX       IARM
780     &       ABAT2',/,I5,7F10.4,'PBAT2       E0        E2        EBAT2      AA
790     &         BB          CC          NP2         NS2          PMXM',/,10F8.4)
800    220   IBATT=(-BB-SQRT(TMP))/(2.*AA)
810 300     EBATT=E0-E1*SS-E2*IBATT
820          PBATT=EBATT*IBATT
830           EBAT2=EBATT*NS2*EBVEH/(NS*EBREF)
840    50     ABAT2=PBAT2/EBAT2
850C
860          IF(SS.EQ.0.0) GO TO 999
870C
880          ABATU=IBATT/NP
890          AMPS=ABS(ABATU)
900          ALIMIT=50./IBASE
910          IF(AMPS.LT.ALIMIT)ABATU=SIGN(ALIMIT,ABATU)
920          IF(ABATU.LT.0.0)GO TO 400
930         IF(IBTYP.GE.2) GO TO 375
940C
950C
960C      COMPUTE MODEL COEFFICIENTS
970C
980 350     ECO=ALPHAT(2)+BETAT(2)*ABATU
990          ELIN=ECO*(1.-SS)+SS*EIT
1000          GAMM=ALPHAT(4)+BETAT(4)*ABATU
1010          GANN=ALPHAT(3)+BETAT(3)*ABATU
1020          EU1=ECO-EIT-(ALPHAT(1)+BETAT(1)*ABATU)*(GANN*SS**(GANN
1030       &-1.0)*(1.0-SS)**GAMM+GAMM*(SS**GANN)*(1.0-SS)**(GAMM-1.0))
1040          EU2=-BETAT(2)*(1.0-SS)-BETAT(1)*(SS**GANN)*(1.-SS)**GAMM
1050       &     -(ALPHAT(1)+BETAT(1)*IBATT)*(BETAT(3)*ALOG(SS)*(SS*GANN)
1060       &     *(1.-SS)**GAMM+BETAT(4)*(ALOG(1.-SS))*((1.-SS)**GAMM)*(SS**
1070       &     GANN))
1080          EBU=ELIN+(ALPHAT(1)+BETAT(1)*ABATU)*(SS*GANN)*(1.0-SS)**GAMM
1090          EOU=EBU+EU1*SS+EU2*ABATU
1100          E0=EOU*NS
1110          E1=EU1*NS
1120          E2=EU2*NS/NP
1130C
1140          GO TO 999
1150 375     PB=PBATT*PBASE/ (NP*NS*NC1)
1160        CUR=IBATT*IBASE/NP
1170         CALL BCOMP(PB,IBTYP,SS,CUR,ECELL,ITER,IB,EB)
1180        E0=ECELL(1)*NS*NC1/VBASE
```

```
      E1=0.0
      E2=ECELL(3)*NS*NC1/(NP*RBASE)
C     IF NO REGENERATION SKIP THIS SECTION
       IF(IBATT.GT.0.0)GO TO 999
400   AX=AR1+AR2*SS+AR3*SS**2.
       ACU=-IBATT*IBASE/UAHC
       SL=BR1+2.*BR2*ACU
      EO=(AX-SL*ACU)*NS/VBB
      E1=0.0
       E2=SL*IBASE*NS/(NP*VBB*UAHC)
410   GO TO 999
C
C     COMPUTE THE BATTERY STATE OF CHARGE(SS), KWH AND AMP-HOURS
C
 500    IF(TDCRG.LE.0.0) TDCRG=DTP
       IF(MON.EQ.0) PBAT2=0.0
       IF(MON.EQ.0) ABAT2=0.0
       AHU2=AHU2+ABAT2*DTP/NP2
        ACUR=AHU2/TDCRG
       DCRATE=UAHC2/ACUR
       ACCX=ACUR*IBASE
      ACCY=ACCX*UAHC/UAHC2
600    CALL LBAT(ACCY,IBTYP,AHC2)
       AHC2=AHC2*UAHC2/UAHC
       TDCRG=TDCRG+DTP
       SS=SS+IBASE*ABAT2*DT/(3600.*AHC2*NP2)
       BCHG=1.0-SS
       ENBAT=ENBAT+ABAT2*EBAT2*PBASE*DT/3.6E6
       IF(ABAT2.GT.0.0)EBOUT=EBOUT+PBAT2*DT*PBASE/3.6E6
       IF(ABAT2.LT.0.0)EBIN=EBIN+PBAT2*DT*PBASE/3.6E6
999    RETURN
       END
C
       SUBROUTINE  LBAT(ACCX,IBTYP,AHC2)
       DIMENSION AH(8,3),DCR(8,3)
       DATA AH/200.,190.,180.,170.,145.,128.,108.,96.,180.,172.,166.
     6,160.,154.,142.,130.,106.,141.8,132.3,122.9,114.8,104.,95.,85.,76./
       DATA DCR/5.0,38.,60.,85.,145.,200.,300.,400.,0.,50.,100.,150.
     6,200.,300.,400.,600.,0.,50.,100.,150.,200.,250.,300.,350./
       J=IBTYP
      IF(ACCX.LE.DCR(8,J)) GO TO 30
C
       WRITE(6,111)ACCX,DCR(8,J)
111    FORMAT(5X,'ACCX EXCEEDS MAX VALUE ',/,5X,'ACCX=',F10.4,'DCMAX
     6=',F10.4)
30     DO 50 I=1,8
       IF(DCR(I,J).GT.ACCX) GO TO 100
50     CONTINUE
100    AHC2=(AH(I,J)-AH(I-1,J))/(DCR(I,J)-DCR(I-1,J))*(ACCX-DCR(I-1,J)
     6)+AH(I-1,J)
999    RETURN
        END
       SUBROUTINE BCOMP (PB,IBTYP,SS,CUR,ECELL,ITER,IB,EB)
C
      PARAMETER NDAT=6
       DIMENSION BCGS(NDAT),AMP(NDAT),ESNF(NDAT,NDAT),ESNZ(NDAT,NDAT)
     6,ECELL(3)
C     COMPUTE CELL CHARACTERISTICS
C
C     COMPUTATION OF BATTERY CHARACTERISTICS FOR
```

```
1790C      NI-FE AND NI-ZN
1800C      INPUTS:  PBALT, S, IBTYP
1810C
1820C      IBTYP=1 PB ACID
1830C      IBTYP=2  NI-FE
1840C      IBTYP=3  NI-ZN
1850C      ROW = S     COL = AMP
1860C      BATTERY DATA
1870        DATA BCGS/0.,.2,.4,.6,.8,.9/,AMP/5.0,100.,150.,200.,250.,350./
1880        DATA ESNF/1.17,1.135,1.105,1.09,1.06,1.03,1.08,1.048,
1890       &           1.02,1.0,0.976,0.94,1.032,1.004,0.976,0.956,
1900       &           .928,0.896,.988,.954,.93,.908,.88,.852,
1910       &           .946,.91,.884,.86,.836,.808,.855,.82,.795,.77,
1920       &           .745,.72/
1930       DATA ESNZ/1.713,1.69,1.675,1.65,1.59,1.53,1.67,1.65,1.63,
1940       &1.6,1.545,1.485,1.64,1.62,1.6,1.57,1.515,1.46,1.61,1.58,
1950       &1.56,1.525,1.475,1.43,1.565,1.537,1.512,1.475,1.425,
1960       &1.385,1.525,1.5,1.465,1.425,1.34,1.3/
1970C      SEARCH FOR REGION IN WHICH DATA IS STORED
1980        IF(SS.LT.0.0.OR.SS.GT.0.90)GOTO555
1990        IF(SS.LT.BCGS(6))GOTO50
2000        II=NDAT-1
2010        GOTO200
2020     50 DO 100 I=1,NDAT
2030        II=I-1
2040        IF(SS.LT.BCGS(I))GOTO200
2050    100 CONTINUE
2060    200 IF(CUR.LT.5.0)CUR=5.0
2070        IF(CUR.GT.1000.)GOTO555
2080        IF(CUR.LT.300.)GOTO400
2090        JJ=NDAT-1
2100        GOTO500
2110    400 DO 450 J=1,NDAT
2120        JJ=J-1
2130        IF(CUR.LT.AMP(J))GOTO500
2140    450 CONTINUE
2150    500 IF(IBTYP.EQ.3)GOTO550
2160        EOSI11=ESNF(II,JJ)
2170        EOSI12=ESNF(II,JJ+1)
2180        EOSI21=ESNF(II+1,JJ)
2190        EOSI22=ESNF(II+1,JJ+1)
2200        GOTO600
2210    550 EOSI11=ESNZ(II,JJ)
2220        EOSI12=ESNZ(II,JJ+1)
2230        EOSI21=ESNZ(II+1,JJ)
2240        EOSI22=ESNZ(II+1,JJ+1)
2250    600 DELS=(SS-BCGS(II))/(BCGS(II+1))-BCGS(II))
2260        EOSI=EOSI11+(EOSI21-EOSI11)*DELS
2270        EOS2=EOSI12+(EOSI22-EOSI12)*DELS
2280        SL=(EOS2-EOS1)/(AMP(JJ+1)-AMP(JJ))
2290        ECELL(1)=EOS1-SL*AMP(JJ)
2300        ECELL(2)=0.0
2310        ECELL(3)=-SL
2320        GOTO999
2330C      ERROR MESSAGE
2340    555 WRITE(6,666)ITER,SS,CUR,PB
2350    666 FORMAT(/,2X,'INPUT DATA OUT OF RANGE,ITER,SS,CUR,PB=',
2360       &/,2X,I5,3E12.5)
2370        STOP
2380    999 RETURN
2390        END
```

```
10      SUBROUTINE ERR1(AA,AB,AC,AD)
20C     INCLUDE HYPR2
100      IF(ITX2.LT.2)WRITE(6,100)T,AA,AB,AC,AD
101        ITX2=ITX2+1
110 100 FORMAT(1H ,'MAX POWER EXCEEDED',5E9.4)
120      AD=100.
130 200 RETURN
140      END


10      SUBROUTINE PFCAL(A1,A2,A3)
20C     INCLUDE HYPR2
100      A3=PF/A1*100.
101        ITZX=ITZX+1
110      IF(A3.GE.100.1 )CALL ERR1(PF,WE,A1,A3)
120 100 RETURN
130      END


10C
20C     ********LINEAR INTERPOLATION ROUTINES********
30C
40      SUBROUTINE LINT(XAR,YAR,TT,VV,AA)
50C     INCLUDE HYPR2
100      DIMENSION XAR(1),YAR(1)
101        IGO=0
102         T2=T
110 10     ITI=0
120      II=XAR(1)
130      DO 100 I=3,II
140      IX=I
150      IF(T2.LE.XAR(I) )GO TO 200
160 100 CONTINUE
170 200 IF((T2+(DT*.9)).GT.XAR(IX).AND.IGO.EQ.0)ITI=1
180      VO=VN
190      VN=YAR(IX-1)+(YAR(IX)-YAR(IX-1))*(T2-XAR(IX-1))/
200     &       (XAR(IX)-XAR(IX-1))
201         IF(IGO.EQ.1) GO TO 222
202         VV=VN
203         IGO=1
204         T2=TT+1
205        GO TO 10
206 222     AA=VN-VO
230      TO=T2
240 999 RETURN
250      END
```

```
10      SUBROUTINE LIN2(XAR,YAR,IXM,XV,YV)
20C     INCLUDE HYPR2
100      DIMENSION XAR(1),YAR(1)
110      DO 100 I=2,IXM
120      IX=I
130      IF(XV.LE.XAR(I))GO TO 200
140 100 CONTINUE
150      WRITE(6,105)XV,XAR(IXM),IX,ISUB,T,ICYC
160 105 FORMAT(1H,'XVAL OUT OF RANGE(LIN2)',2E11.5,2I5,F7.1,I5)
170 200 YV=YAR(IX-1)+(YAR(IX)-YAR(IX-1))*(XV-XAR(IX-1))/(XAR(IX)-
180     &       XAR(IX-1))
190      RETURN
200      END


10      SUBROUTINE BLIN(XAR,YAR,ZAR,IXM,IYM,XV,YV,ZV)
20C     INCLUDE HYPR2
100      DIMENSION XAR(IXM),YAR(IYM),ZAR(IXM,IYM)
110      DO 100 I=2,IXM
120      IX=I
130      IF(XV.LE.XAR(I))GO TO 200
140 100 CONTINUE
150      WRITE(6,105)XV,XAR(IXM),IXM
160 105 FORMAT(1H ,'XVAL OUT OF RANGE (BLIN)',2E11.5,I5)
170 200 DO 210 I=2,IYM
180      IY=I
190      IF(YV.LE.YAR(I))GO TO 300
200 210 CONTINUE
210      WRITE(6,215)XV,YV,YAR(IYM),IY
220 215 FORMAT(1H ,'YVAL OUT OF RANGE (BLIN)',3E11.5,I5)
230 300 XF=(XV-XAR(IX-1))/(XAR(IX)-XAR(IX-1))
240      YF=(YV-YAR(IY-1))/(YAR(IY)-YAR(IY-1))
250      V1=XF*ZAR(IX,IY-1)+ZAR(IX-1,IY-1)*(1.-XF)
260      V2=XF*ZAR(IX,IY)+ZAR(IX-1,IY)*(1.-XF)
270      V3=(V2-V1)*YF+V1
280C     V4=YF*ZAR(IX,IY)+ZAR(IX,IY-1)*(1.-YF)
290C     V5=YF*ZAR(IX-1,IY)+ZAR(IX-1,IY-1)*(1.-YF)
300C     V6=(V4-V5)*XF+V5
310C     ZV=(V6+V3)/2.
320C     IF((V6-V3).GT..001)CALL ERR2
330      ZV=V3
340      RETURN
350      END


10      SUBROUTINE ERR2
20C     INCLUDE HYPR2
100      WRITE(6,100)ISUB
110 100 FORMAT(1H ,' ***** COMPONANT NOT IMPLEMENTED *****','ISUB=',I3)
120      STOP
130      END
```

```
10C
20C     **********CYCEND ROUTINE*************
30C
40      SUBROUTINE CYCEND(ID)
50C     INCLUDE HYPR1
60C     INCLUDE HYPR2
100      IF(ID.GT.190)RETURN
110      OAR(1,ID)= FLOAT(ICYC)
120      OAR(2,ID) = FLOAT(DCTYP)
130      OAR(3,ID) = FLOAT(II)+1.
140      OAR(4,ID) = KM
150C     (LITERS)
160      OAR(5,ID) = FUEL*3.785412/(RHOF*454.5)
170      OAR(6,ID) = BCHG
180      OAR(7,ID) = (BCI - BCHG)*BSEC(BTYP)*MB/CHGREF
190C     ROLL RES
200      OAR(8,ID) = K1
210C     AERO
220      OAR(9,ID) = K2
230C     VEH INERT
240      OAR(10,ID) = K4
250C     RE OUT
260      OAR(11,ID) = EVD
270C     RE IN
280      OAR(12,ID) = PFMA
290C     TOT ENGINE LNG OUTPUT
300      OAR(13,ID) = TEENG
310C     EN EFF
320      IF(FUEL.GT.0001)OAR(14,ID) = TEENG/(FUEL/81.944)*100.
330C     L/100KM
340      OAR(15,ID) = (OAR(5,ID)*100)/KM
350      IF(FUEL.GT.0001)OAR(16,ID) = KM/(FUEL/10581.44)
360      OAR(17,ID) = EBOUT
370      RETURN
380      END
```

```
10C
20C     ******** ACCELERATION ROUTINE **********
30C
40      SUBROUTINE ACCEL
50C     INCLUDE HYPR1
60C     INCLUDE HYPR2
100       ISUB=13
110       VO=0.
120       WIDE=WIDLEA
130       WIDM=WIDLEA
131         SNEW=SACC
140       DO 12 I=1,10W
150       DO 10 II=1,10D
160    10 OAR(II,I)=0.
170    12 CONTINUE
180C      ***** GRATS (4) MUST BE HIGHEST GEAR TO WORK *******
190       OAR(17,53)=WEMX*DWHL/(5.3052*GRATS(4)*GRATS(5))-2.
200       OAR(17,55)=WMMX*DWHL/(5.3052*GRATS(4)*GRATS(5))-2.
210       IF(DTTYP.EQ.1)OAR(17,55)=WMMX*DWHL/(5.3052*GRM*GRATS(5))-2.
220       OAR(17,54)=AMIN1(OAR(17,53),OAR(17,55))
230       DO 8 I=1,3
240     8 OAR(17,52+I)=AMIN1(105.,OAR(17,52+I))
250       VN=0.
260       IESF=0
270       DO 513 IEIF=1,3
280       GO TO (113,213,313),IEIF
290C      ****** MAX ACCEL HEAT ENG ONLY *********
300  113  PFEA=1.
310       PFMA=0.
320       JE=JENG
330       JM=0.
340       IF(PEMX.LT..1)GO TO 513
350       CALL RK
360       GO TO 513
370C      *****ENG+MTR*******
380  213  PFMA=1.
390       JM=JMTR
400       VO=0.
410       IF((PMMX.LT..1).OR.(PEMX.LT..1))GO TO 513
420       CALL RK
430       GO TO 513
440C      *******MTR ONLY********
450  313  PFEA=0.
460       JE=0.
470       VO=0.
480       IF(PMMX.LT..1)GO TO 513
490       CALL RK
500  513  CONTINUE
510       IF(IESF.EQ.1)WRITE(6,1000)
520       DO 523 I=2,50
530  523  OAR(13,I)=FLOAT(I-1)
540       IMGF=ICNNT(1)
550       IF(IMGF.LT.ICNNT(2))IMGF=ICNNT(2)
560       IF(IMGF.LT.ICNNT(3))IMGF=ICNNT(3)
570       IF(IPRTS.GT.0)GO TO 713
580       WRITE(6,1010)
590       DO 710 II=1,IMGF
600  710  WRITE(6,1020)OAR(13,II),(OAR(I,II),I=1,12)
610  713  VN=0.
620       DO 723 I=1,10
```

```
      VN=VN+10.
      OAR(I,52)=VN
      II=ICNNT(1)
      ICYC=1
      T=0.
      IF(OAR(1,II).GT.VN)CALL LIN3(ICYC,II,VN,T)
      OAR(I,53)=T
      ICYC=5
      II=ICNNT(2)
      T=0.
      IF(OAR(5,II).GT.VN)CALL LIN3(ICYC,II,VN,T)
      OAR(I,54)=T
      II=ICNNT(3)
      ICYC=9
      T=0.
      IF(OAR(9,II).GT.VN)CALL LIN3(ICYC,II,VN,T)
      OAR(I,55)=T
  723 CONTINUE
      DO 733 II=53,55
      OAR(11,II)=OAR(5,II)
      OAR(12,II)=OAR(10,II)
      OAR(13,II)=(OAR(6,II)+OAR(5,II))/2.-OAR(3,II)
      OAR(14,II)=OAR(10,II)-OAR(6,II)
  733 CONTINUE
      WRITE(6,1030)
      WRITE(6,1040)(OAR(2,I),I=52,55),(OAR(11,I),I=53,55),(OAR(4,I),
     &      I=52,55),(OAR(12,I),I=53,55),(OAR(6,I),I=52,55),
     &      (OAR(13,I),I=53,55),(OAR(8,I),I=52,55),(OAR(14,I),I=53,55),
     &       (OAR(10,I),I=52,55)
 1000 FORMAT(1H0,'MAX ACCELERATION EXCEEDED DURING THIS RUN')
 1010 FORMAT(/1H0,T15,'HEAT ENGINE ONLY',T55,'ENG + MTR',
     &       T85,'MOTOR ONLY'/1H0,T4,'TIME',T12,'V(KM/HR)',
     &     T22,'A(KM/HR-S)',T33,'WE(RPM)',T43,'WM(RPM)'//)
 1020 FORMAT(1X,T2,F4.1,3(4X,F6.1,4X,F6.1,5X,F6.0,5X,F6.0))
 1030 FORMAT(1H0,T4,'MAX THROTTLE RESULTS:'/1H0,T12,
     &'ENG ONLY  ENG+MTR  MTR ONLY',T50,'ENG',T59,'E+M',T68,'MTR'/
     &1H ,T2,'V(KM/HR)',T12,'TIME(SEC)',T41,'V(K/H)',T50,'TIME(SEC)')
 1040 FORMAT(1X,T2,F5.1,T12,3(F6.1,4X),T41,'0-50',T48,2(F6.1,3X),
     &  F6.1/1H ,T2,F5.1,T12,3(F6.1,4X),T41,'0-100',T48,2(F6.1,3X),
     &  F6.1/1X,T2,F5.1,T12,3(F6.1,4X),T41,'30-55',T48,2(F6.1,3X),
     &  F6.1/1X,T2,F5.1,T12,3(F6.1,4X),T41,'60-100',T48,2(F6.1,3X),
     &   F6.1/1X,T2,F5.1,T12,3(F6.1,4X))
      RETURN
      END
```

```
      SUBROUTINE LIN3(IXL,IDY,XV,YV)
C     INCLUDE HYPR1
C     INCLUDE HYPR2
C
C      ****** IN UPDATE MODE: IDY=1= MAX ENG POW, 2= MX ENG SPD,
C     3= MX MTR POW, 4= MX GEN PWR, 5= MX MTR SPD, 6= MX BAT OUT,
C     7= MX BAT IN, 8= MX PD5, 9= MN PD5, 10= MAX VEH VEL, 11= MX V ACC
C     12=ENG PF 13=ENG WF 14= MTRPF 15=GEN PF 16= MTR WF
C     CALL LIN3(-1,IDUM,VALUE,1.)
C
C     ******IN SHIFTA MODE , CALL LIN3(0,1,1.,1.)
C
      IF(IXL)300,400,
      DO 100 III=2,IDY
      IX=III
      IF(XV.LE.OAR(IXL,III))GO TO 200
  100 CONTINUE
      WRITE(6,105)XV,OAR(IXL,IX),IX
  105 FORMAT(1H ,'(XVAL OUT OF RANGE (LIN3)',2E11.5,I5)
      STOP
  200 YV=OAR(13,IX-1)+(OAR(13,IX)-OAR(13,IX-1))*
     &       (XV-OAR(IXL,IX-1))/(OAR(IXL,IX)-OAR(IXL,IX-1))
      RETURN
  300 OAR(IDY,51)=AMAX1(OAR(IDY,51),XV)
      RETURN
  400 DO 410 I=1,3
      GO TO (420,430,440),DTTYP
  420 SHIFTA(I)=.95*WEMX*DWHL/(5.3052*GRATS(I)*GRATS(5))
      GO TO 410
  430 SHIFTA(I)=AMIN1(WEMX,WMMX)*.95*DWHL/(5.3052*GRATS(I)*GRATS(5))
      GO TO 410
  440 WRITE(6,441)DTTYP,T,IXL,IDY,XV,YV,SHIFTA(1)
  441 FORMAT(1X,I8,E10.5,2I8,3E10.5)
  410 CONTINUE
      RETURN
      END
```

```
10C     ********** RUNGA-KUTTA SOLN ********
20      SUBROUTINE RK
30C     INCLUDE HYPR1
40C     INCLUDE HYPR2
100      ISUB=21
110      DO 225 IMGF=2,50
120      ICNNT(IEIF)=IMGF
130      DO 215 I1=1,10
140      PDS=1.
150 212 CALL PMAXA(V0,1)
160      CALL DVDT(V0)
170      K1=.1*A
180      CALL PMAXA(V0+.5*K1,2)
190      CALL DVDT(V0+.5*K1)
200      K2=.1*A
210      CALL PMAXA(V0+.5*K2,2)
220      CALL DVDT(V0+.5*K2)
230      K3=.1*A
240      CALL PMAXA(V0+K3,2)
250      CALL DVDT(V0+K3)
260      K4=.1*A
270      VN=V0+1./6.*(K1+2.*K2+2.*K3+K4)
280      IF(A.GT.19.5)GO TO 210
290      V0=VN
300      GO TO 215
310 210 PDS=PDS-.05
320      IESF=1
330      GO TO 212
340 215 CONTINUE
350      IKNT=9
360      IF(IEIF.EQ.1)IKNT=1
370      IF(IEIF.EQ.2)IKNT=5
380      OAR(IKNT,IMGF)=VN
390      OAR(IKNT+1,IMGF)=A
400      OAR(IKNT+2,IMGF)=WL
410      OAR(IKNT+3,IMGF)=WM
420      IF(VN.GT.OAR(17,52+IEIF))GO TO 226
430 225 CONTINUE
440 226 RETURN
450      END
```

```
10C
20C     ********** ACCEL EQUATION FOR MAX THROTTLE ROUTINE ******
30C
40      SUBROUTINE OVST(VB)
50C     INCLUDE HYPR1
60C     INCLUDE HYPR2
100      ISUB=20
110C     DIMENSION TQ(11)
120C     IF(IFIX2.EQ.0)GO TO 101
130C     TQ(1)=PMXE*PFEA*PDS*9549.3*NT/WE*GRT*GRATS(5)
140C     TQ(2)=JE*2.*A*GRT*GRATS(5)*NT*ND/(3.6*DWHL)
150C     TQ(3)=PMXM*PFMA*PDS*9549.3*NTMT*ND/WM*GRM*GRATS(5)
160C     TQ(4)=JM*2.*A*GRATS(5)*GRM*NTMT*ND/(3.6*DWHL)
170C     TQ(5)=PACC*9549.3*NT*ND/WE
180C     TQ(6)=JWHL*2.*A/(3.6*DWHL)
190C     TQ(7)=CROL*MV*DWHL*(1.+.911344*CROL1*V+CROL2*(.911344*CROL2)**2)*4.90332
200C     TQ(8)=CD*AF*(VB+VWIND)**2*DWHL/43.2
210C     TQ(9)=S*MV*9.807*DWHL/200.
220C     TQ(10)=MV*A*DWHL/(2.*3.6)
230C     TQ(11)=TG(1)-TQ(2)+TQ(3)-TQ(4)-TQ(5)-TQ(6)-TQ(7)-TQ(8)-TQ(9)
240C     WRITE(22,100)(TQ(I),I=1,11)
250C100  FORMAT(1X,11E10.4)
260C101  CONTINUE
270      A=(GRATS(5)*9549.3*(NT*ND/WE*(PMXE*PFEA*PDS-PACC(1))*GRT+NTMT*
280     &    ND/WM*GRM*PMXM*PFMA*PDS)-S*MV*DWHL*4.903-CROL*MV*DWHL*
290     &      (1.+CROL1*.911344*VB+CROL2*(.911344*VB)**2)*4.903325
300     &      -CD*AF*(VB+VWIND)**2*DWHL/43.2)/((MV*DWHL**2/4.
310     &      +JE*GRT*GRATS(5)*NT*ND+JM*GRM*GRATS(5)*NTMTR*ND+JWHL)/
320     &        (1.8*DWHL))
330      RETURN
340      END
```

```
10C
20C     ***** ROUTINE TO FIND MAX POWER OF ENG+MTR AND SHIFT LOGIC ACCEL=MAXVEL*
30C
40      SUBROUTINE PMAXA(VA,ID2)
50C     INCLUDE HYPR1
60C     INCLUDE HYPR2
100      EON=1
110      MON=1
120      ISUB=12
130       IF(ICVT.EQ.0) GO TO (90,111,90),ID2
133        IF(VA.EQ.0.0) VA=0.0001
150        WDS=VA*GRATS(5)*5.30516/DWHL
152        IF(WDS.EQ.0.0) WDS=0.00001
160        IF(ID2.EQ.1) GO TO (601,610,620) ,IELF
161        IF(ID2.EQ.2) GO TO (605,615,625) ,IELF
170C
180 601   WE=SPER*WEMX
190       GRT=WE*3.1416*DWHL/(VA*GRATS(5)*16.6667)
200       IF(GRT.LT.1./ODR)GRT=1./ODR
210        IF(GRT.GT.RR) GRT=RR
220 605   WE=WDS*GRT
230       NT=ECVT
240       IF(WE.LT.WIDLEA) WE=WIDLEA
241        GO TO 113
250C
260 610   WM=WBASE*WFACT
270       GR1=WM/(WDS )
280       GR2=SPER*WEMX/WDS
290       WE=WEMX*SPER
300       GRT=AMAX1(GR1,GR2)
310       IF(GRT.LT.1./ODR) GRT=1./ODR
320       IF(GRT.GT.RR) GRT=RR
330 615   WM=GRT *WDS
340       WE=WDS*GRT
350        IF(WM.LT.WIDLEA) WM=WIDLEA
360       IF(WE.LT.WIDLEA) WE=WIDLEA
370        NT=ECVT
371          GO TO 113
380C
390 620   WM=WBASE*WFACT
400       GRT=WM/( WDS)
410       IF(GRT.LT.1./ODR)GRT=1./ODR
420       IF(GRT.GT.RR) GRT=RR
430 625   WM=WDS *GRT
440       NT=ECVT
450       GO TO 113
460  90  DO 92 I=1,3
470      IGEAR=I
480  92  IF(VA.LE.SHIFTA(I))GO TO 94
490      IGEAR=4
500  94  NT=NTM(IGEAR)
510      GRT=GRATS(IGEAR)
520 111  WE=VA*16.667*GRT*GRATS(5)/(3.1416*DWHL)
530      IF(IFIL3.EQ.0) GO TO 113
540      WRITE(23,600)A,VA,WE,I1,VO,K1,K2,K3,K4,IMGF
550 600  FORMAT(1X,F8.4,2F10.4,I3,5F8.4,I3)
560 113  GO TO (112,212,312),DTTYP
570 112  IF(ICVT.EQ.0)WM=WE*GRM/GRT
580      NTMT=NTMTR
590      GO TO 512
```

```
600 512 IF(ICVT.EQ.0)WM=WE
610       GRM=GRT
620       NTMT=NT
630       GO TO 512
640 312 CALL ERR2
650 512 GO TO (560,550,570),ID2
660 550 IF(WE.GT.WEMX)WE=WEMX
670       IF(WM.GT.WMMX)WM=WMMX
680 560 WT=WE
681         IF(IEIF.EQ.3) WT=WM
690C      GO TO(562,564,566),IEIF
700C562 CALL ENG(1)
710C      RETURN
720C564 CALL ENG(1)
730C      CALL MTR(1)
740C      RETURN
750C56  CALL MTR(1)
760C      CALL ACC
770C      PACC(1)=PACC(4)
780       CALL ENG(1)
790       CALL MTR(1)
800       IF(IEIF.EQ.3)PACC(1)=PACC(4)
810 570 RETURN
820       END
```

```
10C
20C     ******** MAXIMUM VELOCITY ROUTINE **********
30C
40      SUBROUTINE MAXVEL
50C     INCLUDE HYPR1
60C     INCLUDE HYPR2
100      ISUB=22
110      DO 137 IMON=1,2
120      OAR(1,1)=2.
130      OAR(2,1)=24.
140      IKNT=0
150      IF(IMON.EQ.1)GO TO 106
160      IKNT=3
170      OAR(1,1)=0.
180      OAR(2,1)=0.
190 106  DO 116 IEIF=1,3
200      IFORM2(IKNT+IEIF)=IA3
210      GO TO (136,146,156),IEIF
220 136  PFEA=1.
230      PFMA=0.
240      V=0.
250      IF(PEMX.LT..1)GO TO 225
260      GO TO 166
270 146  PFMA=1.
280      V=0.
290      IF((PEMX.LT..1).OR.(PMMX.LT..1))GO TO 225
300      GO TO 166
310 156  PFEA=0.
320      V=0.
330      IF(PMMX.LT..1)GO TO 225
340      GO TO 166
350 166  DO 176 IEON=1,200
360      V=40.+FLOAT(IEON)
370      CALL PMAXA(V,3)
380      WT=WE
390      GO TO (186,196,206),IEIF
400 186  IF(WE.GT.WEMX)GO TO 226
410      CALL ENG(1)
420      GO TO 216
430 196  IF((WE.GT.WEMX).OR.(WM.GT.WMMX))GO TO 226
440      CALL ENG(1)
450      CALL MTR(1)
460      GO TO 216
470 206  WT=WM
480      IF(WM.GT.WMMX)GO TO 226
490      CALL MTR(1)
500      CALL ACC
510      PACC(1)=PACC(4)
520 216  A=ND*(PMXE*PFEA*NT+PMXM*PFMA*NTMT-PACC(1)*NT)
530     &-OAR(1,1)*MV*V/36708.-CD*AF*(V+OAR(2,1))**2*V/77760.
540     &-CROL*MV*(1.+CROL1*.911344*V+CROL2*(.911344*V)**2)*V/367.0978
550      IF(A.LE.0.)GO TO 225
560 176  CONTINUE
570      WRITE(6,177)A,WE,PMXE,WM,PMXM,PACC,ISUB,IEIF,IMON
580 177  FORMAT(1H ,'MAX VEL NOT ATTAINED',1X,6E10.4,3I5)
590 225  IFORM2(IKNT+IEIF)=IA4
600 226  OAR(IMON+2,IEIF)=V
610 116  CONTINUE
620 137  CONTINUE
630      WRITE(6,1060)(OAR(3,I),I=1,3),(OAR(4,I),I=1,3),
```

```
     &    (IFORM2(I),I=1,6)
1060 FORMAT(1HO,T8,'(2% SLOPE, 24 KM/HR HEADWIND)',T41,'(0% SLOPE,'
     &     ,'0 HEADWIND)'/1X,'MAX VEL',3F8.1,T39,3F8.1/
     &       1X,'LIMIT',T10,3A8,T40,3A8//)
      RETURN
      END
```

```
10C
20C     ***** REINITIALIZATION ROUTINE **********
30C
40C     COMMENTS: MV IS NOT REINITIALIZED- IF NEW MVTOT VIA WEIGHT
50C     PROPAGATION IS DESIRED. INPUT MV=0. FOR NEW SET OF RUNS
60C
70      SUBROUTINE REINT
80C     INCLUDE HYPR1
90C     INCLUDE HYPR2
100      DO 100 I=1,IOW
110      DO 100 II=1OD
120 100 OAR(II,I)=0.
130      ERGEN=0.
140      EHIN=0.
150      EHOUT=0.
160      TMEFF=0.
170      TEFNG=0.
180      EVD=0.
190      FUEL=0.
200      KM=0.
210      TAC=0.
220      IEON=0
230      IMON=0
240      ICYC=1
250      IKNT=1
260      ITER=0
270      IC=1
280      IBCF=0
290      T=0.
300      VN=0.
310      VO=1.E10
320C     MUST BE INPUT AT START OF EACH RUN TO OPERATE
330      IESF=0
340      K1=0.
350      K2=0.
360      K3=0.
370      K4=0.
380      PFEA=0.
390      PFMA=0.
400      PM=0.
410      PE=0.
420      WE=0.
430      WM=0.
440      IINIT=1
450      RETURN
460      END
```

```
100C
101C       ****** SUBROUTINE  DATE8 **************
102C
103C THIS SUBROUTINE GETS CURRENT TIME AND DATE
104C AND PUTS VALUE (ASCII STRING) INTO STRINGS
105C       JDATE    AND      JTIME
106C
107        SUBROUTINE DATE8(JDATE,JTIME)
108        CHARACTER*8 JDATE,JTIME
109C
110        CALL DATIM(JDATE,T)
111        ENCODE(JTIME,10) T
112 10     FORMAT(1X,F6.2,1X)
113C
114        RETURN
115        END
```

```
10C*************************************************
20C*****  FILE DEFINITION SET-UP FOR HYVEC
30C******    SUBROUTINE  ASSIGN (ASSIGN LU FOR OUTPUT)
40C******    ENTRY       FILES  (TO SET UP LOGICAL UNITS)
50C******    ENTRY       FILED  (TO DETACH LOGICAL UNITS)
60C*************************************************
70C
80        SUBROUTINE ASSIGN(LU)
90        CHARACTER FTYPE*8,IFILE*8,FILE*20
95C
100C
105        INTEGER ICBUF(5)/2,1,2000,0,0/
110C OPEN OUTPUT FILE USER SUPPLIES AS LOGICAL UNIT 'LU'
120C THESE ARE THE OUPUT FILES WHEN IFIL,IFIL2,IFIL3 ARE
130C NON-ZERO VALUES.
140C
150        WRITE(6,5) LU
160 5      FORMAT (' GIVE FILENAME FOR LOGICAL UNITS # ',I2)
170        READ(5,15) IFILE
180        ENCODE(FILE,16) IFILE
190        CALL OPENF(LU,FILE,ISTAT,3,0,ICBUF)
200        IF (ISTAT.NE.0) GOTO 200
210        RETURN
220C
230C************************************
240C************************************
250C
260        ENTRY FILES
270C
280C FILENAMES ARE EXPECTED TO BE UNDER THE MAIN ID
290C  NO SUBCATALOG STRUCTURE EXPECTED
300C
310        FTYPE='INPUT1'
320        WRITE(6,10) FTYPE
330 10     FORMAT(' GIVE FILENAME FOR ',A8,'? ')
340        READ(5,15) IFILE
350 15     FORMAT(A6)
360C
370C CHECK IF INPUT FILE GIVEN EXISTS
380C
390        LU=1
400        ENCODE(FILE,16) IFILE
410 16     FORMAT(A8,1H;)
420        CALL OPENF (LU,FILE,ISTAT,1,0,1)
430        IF (ISTAT.NE.0) GO TO 300
440        CALL DETACH (LU,ISTAT,)
450        IF (ISTAT.NE.0) GO TO 600
460C
470C FOR ASCBCD FUNCTION, THE FILENAME MUST BE FOLLOWED
480C BY A SEMI-COLON DELIMETER FOLLOWED BY DESTINATION FILE
490C
500        ENCODE(FILE,17) IFILE
510 17     FORMAT('ASCB ',A8,';TEMP1\')
520        CALL CALLSS (FILE)
530C
540C OPEN LU#15 AS READ ONLY, MODE AS REQUIRED, AND DO NOT
550C CREATE FILE IF IT DOESN'T EXIST
560C
570        CALL OPENF(15,'TEMP1;',ISTAT,1,0,1)
580        IF (ISTAT.NE.0) GOTO 200
```

```
590C
600C INPUT FILE (LU#15) OPENED OK, SO CONTINUE
610C
620C
630         FTYPE='INPUT2'
640         WRITE(6,10) FTYPE
650         READ(5,15) IFILE
660C
670C CHECK IF INPUT FILE GIVEN EXISTS
680C
690         LU=1
700         ENCODE(FILE,16) IFILE
710         CALL OPENF (LU,FILE,ISTAT,1,0,1)
720         IF (ISTAT.NE.0) GO TO 300
730         CALL DETACH (LU,ISTAT,)
740         IF (ISTAT.NE.0) GO TO 600
750C
760C FOR ASCBCD FUNCTION, THE FILENAME MUST BE FOLLOWED
770C BY A SEMI-COLON DELIMETER FOLLOWED BY DESTINATION FILE
780C
790         ENCODE(FILE,19) IFILE
800 19      FORMAT('ASCB ',A8,';TEMP2\')
810         CALL CALLSS (FILE)
820C
830C OPEN LU#17 AS READ ONLY, MODE AS REQUIRED, AND DO NOT
840C CREATE FILE IF IT DOESN'T EXIST
850C
860         CALL OPENF(17,'TEMP2;',ISTAT,1,0,1)
870         IF (ISTAT.NE.0) GOTO 200
880C
890C INPUT FILE (LU#17) OPENED OK, SO CONTINUE
900C
910C
920C OPEN LU#25 AS READ ONLY, ETC.  THIS FILE CONTAINS
930C THE EPA DATA VALUES
940C
950         IFILE='EPADATA'
960         CALL OPENF (25,'/SHYVEC/EPADATA;',ISTAT,1,0,1)
970         IF (ISTAT.NE.0) GOTO 200
980 100     CONTINUE
990         RETURN
1000C
1010C**************************************
1020C**************************************
1030C
1040          ENTRY FILED
1050C
1060C NOW DETACH ANY AND ALL LOGICAL UNITS WHICH WERE
1070C ATTACHED TO FILENAMES
1080C
1090         LU=15
1100         CALL DETACH (LU,ISTAT,)
1110         IF (ISTAT.NE.0) GOTO 600
1120         LU=25
1130         CALL DETACH (LU,ISTAT,)
1140         IF (ISTAT.NE.0) GOTO 600
1150         LU=21
1160         CALL DETACH (LU,ISTAT,)
1170         IF (ISTAT.NE.0) GOTO 600
1180         LU=22
```

```
      CALL DETACH (LU,ISTAT,)
      IF (ISTAT.NE.0) GOTO 600
      LU=23
      CALL DETACH (LU,ISTAT,)
      IF (ISTAT.NE.0) GOTO 600
      LU=18
      CALL DETACH (LU,ISTAT,)
      IF (ISTAT.NE.0) GOTO 600
      LU=17
      CALL DETACH (LU,ISTAT,)
      IF (ISTAT.NE.0) GOTO 600
      LU=23
      CALL DETACH (LU,ISTAT,)
      IF (ISTAT.NE.0) GOTO 600
500   RETURN
C
C * * * * E R R O R   M E S S A G E S * * * *
C
C
C ERROR IN OPENING THE FILE --PRINT MESSAGE AND GIVE
C ISTAT VALUE FOR ERROR AND THE FILENAME
C DESCRIPTION GIVEN PG 14-12 IN USERS MANUAL FOR ISTAT
C
200   CONTINUE
      WRITE(6,201) ISTAT,IFILE
201   FORMAT(10X,'***ERROR*** IN OPENF,  ISTAT,FILE=',I5,1H,,A8)
      STOP
C
C ERROR IN ATTACHING THE FILE TO CHECK FOR FILE'S EXISTENCE
C
300   CONTINUE
      WRITE(6,301) ISTAT,IFILE
301   FORMAT(10X,'***ERROR*** IN ATTACH,  ISTAT,FILE=',I5,1H,,A8)
      STOP
C
C ERROR IN DETACH PROCEDURE--PRINT MESSAGE
C
600   WRITE(6,601) ISTAT,LU
601   FORMAT(10X,'***ERROR*** IN DETACH,  ISTAT,LU=',I5,1H,,I5)
      STOP
      END
```

```
                                                ***********************
020         SUBROUTINE BAT(IPBAT)
030C ***********************************************************
040C
050C       THIS SUBROUTINE IS USED TO COMPUTE
060C            (1) THE BATTERY CURRENT + VOLTAGE FOR A GIVEN BATTERY POWER
070C            (2) THE PARAMETERS (E0,E1,E2) OF THE SIMPLIFIED BATTERY MODEL
080C                GIVEN BY EBATT = E0-E1*55-E2*IBATT
090C
091C       INCLUDE HYPR1
092C       INCLUDE HYPR2
100C
110        DIMENSION ECELL(3)
120         DATA AR1,AR2,AR3,BR1,BR2/6.556,0.1,-.714,.352,.127/
130C       IBATT = TOTAL BATTERY CURRENT
140C       ABATU = UNIT BATTERY CURRENT
150C       PBATT = TOTAL BATTERY POWER
160C       EBATT = TOTAL BATTERY VOLTAGE
170C
180C      IF HERE FOR THE FIRST TIME  INITIALIZE
190         IF(ITR.NE.1) GO TO 101
200C
210C       CONVERT BATTERY PARAMETERS
220C
230         RBASE=VBASE/IBASE
240        VNZ=VCNZ/VBASE
250        VNF=VCNF/VBASE
260        RNF=RCNF/RBASE
270        RNZ=RCNZ/RBASE
280         VBR=6.3*VBASE/EBREF
290         ALPHAT(1)=ALPHA(1)/VBASE
300         ALPHAT(2)=ALPHA(2)/VBASE
310         ALPHAT(3)=ALPHA(3)
320         ALPHAT(4)=ALPHA(4)
330         ALPHAT(5)=ALPHA(5)/VBASE
340         BETAT(1)=BETA(1)/RBASE
350         BETAT(2)=BETA(2)/RBASE
360         BETAT(3)=BETA(3)*IBASE
370         BETAT(4)=BETA(4)*IBASE
380         BETAT(5)=BETA(5)/RBASE
390         EBVE2=EBVEH/VBASE
400         EIT=EII/VBASE
410C      SET INITIAL VALUES
420         SS = 0.0
430         GO TO (10,20,30),IPTYP
440 10      E0 = ALPHAT(2)*NS
450         E1 = (ALPHAT(2)-EIT)*NS
460         E2 = -BETAT(2)*NS/NP
470        GO TO 100
480   20    E0=VNF*NC1*NS
490        E1=0.0
500        E2=RNF*NC1*NS/NP
510        GO TO 100
520     30  E0=VNZ*NC1*NS
530         E1=0.0
540         E2=RNZ*NC1*NS/NP
550 100     A1=AR1*EBREF/(VBASE*ALPHA(2))
560        A2=AR2*EBREF/(VBASE*ALPHA(2))
570        A3=AR3*EBREF/(VBASE*ALPHA(2))
```

```
580       B1=BR1*EBREF/(VBASE*ALPHA(2))
590       B2 =BR2*EBREF/(VBASE*ALPHA(2))
600       B3=BR3*EBREF/(VBASE*ALPHA(2))
610         ITBAT=0
620C      IF BATTERY VOLTAGE IS KNOWN SKIP THIS SECTION
630 101   IF(IBBAT-1)300,200,500
640C
650C      SOLVE QUADRATIC EQUATION TO GET IBATT
660C
670   200 AA=E2
680         BB=E1*SS-E0
690         CC=PBAT2*NC1*NS*NP*UAHC/(NP2*NC2*NS2*UAHC2)
700       TMP=BB*BB-4.*AA*CC
710       IF(TMP.GE.0.0) GO TO 220
720       TMP=0.0
730       ITBAT=ITBAT+1
740         IF(ITBAT.GT.1) GO  TO 220
750         WRITE(6,229) ITER,PM,WM,PE,WE,FLUX,IARM,ABAT2,PBAT2,E0,E2,EBAT2
760      6,AA,BB,CC,NP2,NS2,PMXM
770 229   FORMAT('ITER       PM          WM          PE          WE     FLUX       IARM
780     6       ABAT2',/,15,7F10.4,'PBAT2       E0       E2       EBAT2      AA
790     6         BB          CC          NP2       NS2          PMXM',/,10F8.4)
800   220   IBATT=(-BB-SQRT(TMP))/(2.*AA)
810 300     EBATT=E0-E1*SS-E2*IBATT
820           PBATT=EBATT*IBATT
830            EBAT2=EBATT*NS2*EBVEH/(NS*EBREF)
840    50     ABAT2=PBAT2/EBAT2
850C
860         IF(SS.EQ.0.0) GO TO 999
870C
880         ABATU=IBATT/NP
890         AMPS=ABS(ABATU)
900         ALIMIT=50./IBASE
910         IF(AMPS.LT.ALIMIT)ABATU=SIGN(ALIMIT,ABATU)
920         IF(ABATU.LT.0.0)GO TO 400
930       IF(IBTYP.GE.2) GO TO 375
940C
950C
960C      COMPUTE MODEL COEFFICIENTS
970C
980 350    EC0=ALPHAT(2)+BETAT(2)*ABATU
990         ELIN=EC0*(1.-SS)+SS*EIT
1000         GAMM=ALPHAT(4)+BETAT(4)*ABATU
1010         GANN=ALPHAT(3)+BETAT(3)*ABATU
1020         EU1=EC0-EIT-(ALPHAT(1)+BETAT(1)*ABATU)*(GANN*SS**(GANN
103.      6-1.0)*(1.0-SS)**GAMM+GAMM*(SS**GANN)*(1.0-SS)**(GAMM-1.0))
1040         EU2=-BETAT(2)*(1.0-SS)-BETAT(1)*(SS**GANN)*(1.-SS)**GAMM
1050      &     -(ALPHAT(1)+BETAT(1)*IBATT)*(BETAT(3)*ALOG(SS)*(SS*GANN)
1060      &     *(1.-SS)**GAMM+BETAT(4)*(ALOG(1.-SS))*((1.-SS)**GAMM)*(SS**
1070      &     GANN))
1080         EBU=ELIN+(ALPHAT(1)+BETAT(1)*ABATU)*(SS*GANN)*(1.0-SS)**GAMM
1090         EOU=EBU+EU1*SS+EU2*ABATU
1100         E0=EOU*NS
1110         E1=EU1*NS
1120         E2=EU2*NS/NP
1130C
1140         GO TO 999
1150 375    PB=PBATT*PBASE/ (NP*NS*NC1)
1160       CUR=IBATT*IBASE/NP
1170         CALL BCOMP(PB,IBTYP,SS,CUR,ECELL,ITER,IB,EB)
```

```
1180        E0=ECELL(1)*N5*NC1/VBASE
1190        E1=0.0
1200        E2=ECELL(3)*N5*NC1/(NP*VBASE)
1210C       IF NO REGENERATION SKIP THIS SECTION
1220         IF(IBATT.GT.0.0)GO TO 999
1230  400   AX=AR1+AR2*SS+AR3*SS**2.
1240         ACU=-IBATT*IBASE/UAHC
1250         SL=BR1+2.*BR2*ACU
1260        E0=(AX-SL*ACU)*N5/VBB
1270        E1=0.0
1280         E2=SL*IBASE*N5/(NP*VBB*UAHC)
1290  410   GO TO 999
1300C
1310C       COMPUTE THE BATTERY STATE OF CHARGE(SS), WKWH AND AMP-HOURS
1320C
1330  500     IF(TDCRG.LE.0.0) TDCRG=DTP
1340         IF(MON.EQ.0) PBAT2=0.0
1350         IF(MON.LQ.0) ABAT2=0.0
1360         AHU2=AHC2+ABAT2*DTP/NP2
1370          ACUR=AHU2/TDCRG
1380         DCRATE=UAHC2/ACUR
1390          ACCX=ACUR*IBASE
1400        ACCY=ACCX*UAHC/UAHC2
1410  600    CALL LBAT(ACCY,IBTYP,AHC2)
1420         AHC2=AHC2*UAHC2/UAHC
1430         TDCRG=TDCRG+DTP
1440         SS=SS+IBASE*ABAT2*DT/(3600.*AHC2*NP2)
1450         BCHG=1.0-SS
1460         EBBAT=EBBAT+ABAT2*EBAT2*PBASE*DT/3.6E6
1470         IF(ABAT2.GT.0.0)EBOUT=EBOUT+PBAT2*DT*PBASE/3.6E6
1480         IF(ABAT2.LT.0.0)EBIN=EBIN+PBAT2*DT*PBASE/3.6E6
1490  999    RETURN
1500         END
1510C
1520         SUBROUTINE  LBAT(ACCX,IBTYP,AHC2)
1530         DIMENSION AH(8,3),DCR(8,3)
1540         DATA AH/200.,190.,180.,170.,145.,128.,108.,96.,180.,172.,166.
1550        &,160.,154.,142.,130.,106.,141.6,132.3,122.9,114.8,104.,95.,85.,76./
1560         DATA DCR/5.0,3R.,60.,85.,145.,200.,300.,400.,0.,50.,100.,150.
1570        &,200.,300.,400.,600.,0.,50.,100.,150.,200.,250.,300.,350./
1580         J=IBTYP
1590        IF(ACCX.LE.DCR(R,J)) GO TO 30
1600
1610         WRITE(6,111)ACCX,DCR(R,J)
1620  111    FORMAT(5X,'ACCX EXCEEDS MAX VALUE ',/,5X,'ACCX=',F10.4,'DCMAX
1630        &=',F10.4)
1640   30    DO 50 I=1,8
1650         IF(DCR(I,J).GT.ACCX) GO TO 100
1660   50    CONTINUE
1670  100    AHC2=(AH(I,J)-AH(I-1,J))/(DCR(I,J)-DCR(I-1,J))*(ACCX-DCR(I-1,J)
1680        &)+AH(I-1,J)
1690  999    RETURN
1700          END
1710         SUBROUTINE ICOMP (PH,IBTYP,SS,CUR,ECELL,ITER,IB,EB)
1720C
1730        PARAMETER NDAT=6
1740         DIMENSION BCGS(NDAT),AMP(NDAT),ESNE(NDAT,NDAT),ESNZ(NDAT,NDAT)
1750       &,ECELL(3)
1760C       COMPUTE CELL CHARACTERISTICS
1770C
```

```
1780C      COMPUTATION OF BATTERY CHARACTERISTICS FOR
1790C      NI-FE AND NI-ZN
1800C      INPUTS:  PBALT, S, IBTYP
1810C
1820C      IBTYP=1 PB ACID
1830C      IBTYP=2  NI-FE
1840C      IBTYP=3  NI-ZN
1850C      ROW = S     COL = AMP
1860C      BATTERY DATA
1870        DATA BCGS/0.,.2,.4,.6,.8,.9/,AMP/5.0,100.,150.,200.,250.,350./
1880        DATA ESNF/1.17,1.135,1.105,1.09,1.06,1.03,1.08,1.048,
1890       &           1.02,1.0,0.976,0.94,1.032,1.004,0.976,0.956,
1900       &           .928,0.896,.988,.954,.93,.908,.88,.852,
1910       &           .946,.91,.884,.86,.836,.808,.855,.82,.795,.77,
1920       &           .745,.72/
1930       DATA ESNZ/1.713,1.69,1.675,1.65,1.59,1.53,1.67,1.65,1.63,
1940       &1.6,1.545,1.485,1.64,1.62,1.6,1.57,1.515,1.46,1.61,1.58,
1950       &1.56,1.525,1.475,1.43,1.565,1.537,1.512,1.475,1.425,
1960       &1.385,1.525,1.5,1.465,1.425,1.34,1.3/
1970C      SEARCH FOR REGION IN WHICH DATA IS STORED
1980         IF(SS.LT.0.0.OR.SS.GT.0.90)GOTO555
1990         IF(SS.LT.BCGS(6))GOTO50
2000         II=NDAT-1
2010         GOTO200
2020       50 DO 100 I=1,NDAT
2030         II=I-1
2040         IF(SS.LT.BCGS(I))GOTO200
2050      100 CONTINUE
2060       200 IF(CUR.LT.5.0)CUR=5.0
2070         IF(CUR.GT.1000.)GOTO555
2080         IF(CUR.LT.300.)GOTO400
2090         JJ=NDAT-1
2100         GOTO500
2110       400 DO 450 J=1,NDAT
2120         JJ=J-1
2130         IF(CUR.LT.AMP(J))GOTO500
2140      450 CONTINUE
2150       500 IF(IBTYP.EQ.3)GOTO550
2160         EOSI11=ESNF(II,JJ)
2170         EOSI12=ESNF(II,JJ+1)
2180         EOSI21=ESNF(II+1,JJ)
2190         EOSI22=ESNF(II+1,JJ+1)
2200         GOTO600
2210       550 EOSI11=ESNZ(II,JJ)
2220         EOSI12=ESNZ(II,JJ+1)
2230         EOSI21=ESNZ(II+1,JJ)
2240         EOSI22=ESNZ(II+1,JJ+1)
2250       600 DELS=(SS-BCGS(II))/(BCGS(II+1)-BCGS(II)
2260         EOS1=EOSI11+(EOSI21-EOSI11)*DELS
2270         EOS2=EOSI12+(EOSI22-EOSI12)*DELS
2280         SL=(EOS2-EOS1)/(AMP(JJ+1)-AMP(JJ))
2290         ECELL(1)=EOS1-SL*AMP(JJ)
2300         ECELL(2)=0.0
2310         ECELL(3)=-SL
2320         GOTO999
2330C       ERROR MESSAGE
2340       555 WRITE(6,666)ITER,SS,CUR,PP
2350       666 FORMAT(/,2X,'INPUT DATA OUT OF RANGE,ITER,SS,CUR,PP=',
2360        &/,2X,I5,3E12.5)
2370         STOP
2380       999 RETURN
2390         END
```

```
C
C     ****** OUTPUT ROUTINE **********
C
      SUBROUTINE OUTPUT
      INCLUDE HYPR1
      INCLUDE HYPR2
        DIMENSION TEMPAX(6,200)
       IF(IACCEL.EQ.2)GO TO 494
       IF(IINIT.EQ.0)GO TO 90
       IINIT=0
       CALL DATE8(JDATE,JTIME)
       WRITE(6,500)JDATE,JTIME,DT,DPRT,DCTYP,NCYC,DIST,BCVAL,VMODE,
     &     VEON,VECMD
       WRITE(6,510)CTYP,ETYP,MTYP,IBTYP,REGEN,IFCON,DTTYP,IDOWN,ACTYP,
     &     IFAN,IPS,IAC,IACCEL,IPRTS
       WRITE(6,520)CROL,CROL1,CROL2,KP,PFE,PFM,PFFW,TFWP,PEMX,PMMX,PRAT
       WRITE(6,530)WEMX,WMMX,WFWMX,ESP,MSP,CSP,KOL,JWHL,JENG,JMTR,MFB
       WRITE(6,540)BSE,HSP,MB,ESBMX,PSBMX,MCHN,MVCH,KMP,MPL
       WRITE(6,550)MV,DWHL,MFPT,NCI,BCMN,BCMX,(GRATS(I),I=1,5)
       WRITE(6,570)(NTM(I),I=1,4),AD,GRM,NTPTR,(SHIFT(I),I=1,3),CD,AF
       WRITE(6,560)(SHIFTA(I),I=1,3),FIDLE(ETYP),WIDLE(ETYP),VWIND,S,EPCP
     & ,PBCMX,PRDCMX
          WRITE(6,111)NS,NS2,NP,NP2,NC1,NC2,FBREF,EBVEH,UAHC,UAHC2
  111    FORMAT(/,2X,'NS    NS2   NP    NP2   NC1   NC2   EB1
     &'EB2       UAHC        UAHC2',/,6(F5.0),4(F6.2,4X)/)
        WRITE(6,575)(ITRIP(I),I=1,7),(SSVEL(I),TSS(I),I=1,2),BCEFF,CHGREF
   90  IF(IPRTS.GT.0)GO TO 100
       IF((T+DT/4.)-PRT)100,,
       IF(IFIL2.EQ.0)GO TO 80
       WRITE(22,590)T,V,K1,K2,K4,PWHL,BSFC,PDS,PACC(I),PJE,PJM,PT,
     &       GRT,PE,WDS,PM,WT
   80  PRT=PRT+DPRT
       OAR(1,IC)=T
       OAR(2,IC)=V
       OAR(3,IC)=A
       OAR(4,IC)=PT
       OAR(5,IC)=FLOAT(EON)
       OAR(6,IC)=WE
       OAR(7,IC)=PE
       OAR(8,IC)=EENG
       OAR(9,IC)=FCONS/81.944
       OAR(10,IC)=PM
       OAR(11,IC)=WM
       OAR(12,IC)=EM
       OAR(13,IC)=FLOAT(IMGF)
       OAR(14,IC)=FLOAT(IBCF)
       OAR(15,IC)=PDS
       OAR(16,IC)=EBAT
       OAR(17,IC)=BCHG
       IC=IC+1
       IF(IC.LT.191)GO TO 100
       IC=1
       WRITE(6,620)
       WRITE(6,580)((OAR(I,II),I=1,10),II=1,190)
       IF(IFIL.EQ.0)GO TO 100
       WRITE(21,620)
       WRITE(21,580)((OAR(I,II),I=1,10),II=1,190)
  100  T=T+DT
       ITER=ITER+1
```


C 2

```
      IF(ITI.EQ.1)T=T-0.9*DT
      IX=(T+.0005)/.001
      T=IX*.001
      DTP=DT
      IF(ITI.EQ.1)DTP=DT/10.
      TAC=TAC+DTP
      IF(T.LE.TMIN)RETURN
C       INITIALIZE EGMCO,EGMHC, ETC. AT BEGINNING OF EACH CYCLE
C      AT THE END OF EACH CYCLE
        DISCYC=(KM )/1.609
        EGMCO=EMCO/DISCYC
       EGMCS=EMCS/DISCYC
        EGMHC=EMHC/DISCYC
        EGMNOX=EMNOX/DISCYC
  333 DISAVE=KM
C      COMPUTE TOTAL EMISSION FOR COMPLETE RUN
        EGMHCT=EGMHCT+EGMHC
        EGMCST=EGMCST+EGMCS
        EGMCOT=EGMCOT+EGMCO
  222 EGMNOXT=EGMNOXT+EGMNOX
        ACCX=ACUR*IBASE1
        TEMPAX(1,ICYC)=EGMCO
        TEMPAX(2,ICYC)=EGMHC
        TEMPAX(3,ICYC)=EGMNOX
        TEMPAX(4,ICYC)=EGMCS
        TEMPAX(5,ICYC)=ACCX
        TEMPAX(6,ICYC)=AHC2
  444   FORMAT(/,2X,'EGMCO      EGMHC      EGMNOX     EGMCS      AVCUR
     &'AHC2       ',/,4(F8.4,2X),2(F8.2,2X)/)
C
  140 IF(IPRTS.EQ.2)CALL CYCEND(ICYC)
        MPG=KM*2.408/DAR(5,ICYC)
        FEHE=TEEMG/(TEEMG+EBOUT)
        FGS=1.-22./MPG
        IF(ETYP.EQ.9) TEG=36.6/MPG
        IF(ETYP.EQ.10) TEG=42.2/MPG
        TEM=3.8*SS*AHC2*NS2*NP2*EEVEH/(621.504*KM)
        TKWH=TEM+TEG
         WRITE(6,777) ICYC,MPG,SS,FEHE,FGS,TKWH
  777    FORMAT(/,'ICYC       MPG     SS       FEHE       FGS       TKWH',/,I5,5(
     &  F9.4 )/)
      IF(ITRIP(1).LT.2)GO TO 149
      II=II+1
      IF((NCYC.LE.-1).AND.(KM.GE.DIST))GO TO 200
      IF(II.LT.ITRNUM(IKNT))GO TO 152
      IF(IPRTS.EQ.3)CALL CYCEND(IKNT)
      IKNT=IKNT+1
      IF(IKNT.GT.ITRIP(1))GO TO 200
      DCTYP=ITRIP(IKNT)
      II=0
      GO TO 152
  149 IF(NCYC.GT.0)GO TO 150
      IF(KM.GE.DIST)GO TO 200
  150 IF(ICYC.EQ.NCYC)GO TO 200
        IF(CTYP.NE.4) GO TO 152
        IF(DCHG.GT.BCMD) GO TO 152
        NCYC=ICYC
       GO TO 200
  152 ICYC=ICYC+1
C      AT THE BEGINNING OF EACH CYCLE
```

```
1220C
1230        T=0.
1240        PRT=0.
1250        VH=0.
1260        TO=1.E10
1270        RETURN
1280  200 IF(IPRTS.EQ.0)GO TO 450
1290        IF(IPRTS-2)480,460,470
1300  460 III=ICYC
1310        GO TO 475
1320  470 III=IKNT-1
1330  475 DO 476 II=1,III
1340        WRITE(6,585)(OAR(I,II),I=1,IOD)
1350  476     WRITE(6,444)(TEMPAX(I,II),I=1,6)
1360        GO TO 490
1370C       HEADER
1380  450 WRITE(6,620)
1390        IC=IC-1
1400        WRITE(6,580)((OAR(I,II),I=1,IOD),II=1,IC)
1410  480 IF(IFIL.EQ.0)GO TO 490
1420        WRITE(21,580)((OAR(I,II),I=1,IOD),II=1,IC)
1430        END FILE 21
1440  490 DO 493 I=1,IOD
1450  493 OAR(I,1)=0.
1460        V2=0.
1470        V6=0.
1480        TMEFF=0.
1490        OAR(8,1)=FUEL/(454.5*RHOF)
1500        IF(OAR(8,1).GT.0.)OAR(9,1)=KM/(OAR(8,1)*3.785412)
1510C       KM TO MILES
1520        OAR(6,1)=KM/1.60934
1530        IF(OAR(8,1).GT.0.)OAR(10,1)=OAR(6,1)/OAR(8,1)
1540        OAR(7,1)=FUEL/81.944
1550        V1=(BCI-BCHG)*BSEC(BTYP)*MB
1560        IF(V1.  .0.)V2=V1/(BSPC(BTYP)*MB)
1570        V1=V1/C  GREF
1580        OAR(3,1)=V1+OAR(7,1)
1590        V5=EVD/OAR(3,1)*100.
1600        V3=OAR(7,1)/OAR(3,1)*100.
1610        V4=V1/OAR(3,1)*100.
1620        IF(V1.GT.0.)V6=OAR(6,1)/V1
1630        IF(V1.GT.0.)OAR(1,1)=KM/V1
1640        OAR(2,1)=OAR(6,1)/OAR(3,1)
1650        OAR(4,1)=KM/OAR(3,1)
1660        OAR(5,1)=OAR(6,1)/(OAR(3,1)/34.1123)
1670        IF(OAR(7,1).GT.0.)OAR(12,1)=TEENG/OAR(7,1)*100.
1680C       MTR EFF
1690        IF(OAR(3,193).GT.0.)TMEFF=OAR(1,193)*100./OAR(3,193)
1700C       GEN EFF
1710        IF(OAR(2,193).GT.0.)OAR(4,193)=OAR(4,193)*100./OAR(2,193)
1720C       ENG AVE PF
1730        IF(IEON.GT.0)OAR(5,193)=OAR(5,193)*100./FLOAT(IEON)
1740C       ENG AVE SPD FRAC WF
1750        IF(IEON.GT.0)OAR(6,193)=OAR(6,193)*100./FLOAT(IEON)
1760        IF(IMON.GT.0)OAR(11,1)=EBOUT/(-EBIN+V1)*100.
1770        IFORM=IA1
1780        IF(IPRTS.GT.0)IFORM=IA2
1790        OAR(13,1)=(FUEL*378.5412/2795.33)/KM
1800        WRITE(6,600)IFORM,TAC,ICYC,OAR(6,1),KM,OAR(7,1),V1,OAR(3,1),EVD
1810        WRITE(6,630)V5,V3,V4,OAR(10,1),OAR(9,1),OAR(13,1),V6,OAR(1 1)
```

```
      WRITE(6,640)OAR(2,1),OAR(4,1),OAR(5,1),FBOUT,FBIN,K1,K2,FRGEN
      WRITE(6,650)PFMA,TEFFG,FCHG,BCI,V2,OAR(12,1),TMEFF,OAR(11,1),
     &     OAR(4,193),OAR(2,192)
        WRITE(6,444)EGMCOT,EGMHCT,EGMNOXT,ACCX,AHC2
      IESF=IFIX(OAR(1,192))
      WRITE(6,670)(OAR(I,191),I=1,11),IESF
      WRITE(6,680)(OAR(I,191),I=12,16),ITER,IMON,IEON,(OAR(I,193),I=5,6)
      IF(IACCEL.EQ.0)GO TO 495
  494 CALL ACCEL
      CALL MAXVEL
  495 WRITE(6,660)
      READ(5,665)IFORM2(1)
  665 FORMAT(A1)
      IRUN2=0
      IF(IFORM2(1).EQ.'Y')GO TO 497
      ISTOP=1
      RETURN
  497 IRUN2=1
      CALL REINT
C****FOR USE ON THE CRD 605 COMPUTER, FORMAT STATEMENT #500
C****HAS BEEN CHANGED FROM
C****....A6,A7....
C****TO    ....A4,A1,2A4....
C****FOR PRINTOUT OF DATE ON FILE #6
  500 FORMAT(1H0,T13,'VEHICLE AND RUN INFORMATION  HYVEC'/
     &     1H0,' DATE   TXOT    DT    DPRT  DCTYP NCYCS  DIST',
     &     ' BCVAL VMODE  VEON  VBCMN'/1X,A5,A8,F6.1,F8.1,I4,I6,
     &     F8.1,F5.2,2F6.1,F7.1)
  510 FORMAT(1H0,'CTYP ETYP MTYP BTYP RGEN IECON DTTYP IDWN ACTYP',
     &     ' IFAN IPS AC IACEL IPRTS'/1X,I3,5I5,4I6,1X,2I3,I5,I6)
  520 FORMAT(1H0,T3,'CROL  CROL1  CROL2',T27,'KP   PFE   PFM  PFFW',
     &     '  TFWP  PEMX  PMMX   PRAT'/1X,F6.4,2E8.2,F6.4,2F6.3,F5.2,
     &     F6.1,2F7.1,F7.3)
  530 FORMAT(1H0,' WEMX   WMMX',T18,'WFWMX',T28,'ESP  MSP CSP  KOL',
     &     '  JWHL JENG JMTR',T66,'MFB'/1X,F6.0,F7.0,F10.3,1X,3F5.3,
     &     F4.1,1X,3F5.2,F9.4)
  540 FORMAT(1H0,T5,'BSE',T14,'BSP',T24,'NB',T30,'ESHMX  PSHMX    ',
     &     'MCHN    MCHA   KMP    MPL'/1X,2F9.3,F8.1,F8.2,F7.1,
     &     2F8.1,F4.2,F9.1)
  550 FORMAT(1H0,1X,'MVTOT    DWHL MFPT  BCI   BCMN  BCMX  TRANS ',
     &     'GRATS(1,2,3,4)',T67,'DIFF'/1X,F7.1,F7.3,F5.2,3F6.2,2X,
     &     4F6.3,F8.3)
  560 FORMAT(1H0,' SHIFT POINTS(ACCEL)  ID EF  ID SPD VWIND GRADE',
     &     ' EPCP  PHCMX  PBDCMX'/1X,3F6.1,F8.2,F8.0,F6.1,F7.2,F5.2,F6.1,F8.1)
  570 FORMAT(1H0,' TRANS EFF(1-4)',T25,'DEFF   GRM  MTEFF  SHIFT ',
     &     'POINTS',T64,'CD   AF'/1X,4F5.2,F6.2,F7.3,F6.2,1X,3F6.1,
     &     F7.3,F6.1)
  575 FORMAT(1H0,'# SFTS  DRIVING CYCLES (#DC,TYPE)',T38,'SSV1  TS51',
     &     T50,'SSV2 TS52 BCEFF CHGREF'/1X,I3,I6,5I5,T37,2(F6.1,F6.0),
     &     F5.2,F6.2)
  580 FORMAT(1X,F8.2,2F7.1,E11.4,F3.0,F8.1,E11.4,F7.3,2E11.4,
     &     F8.1,F7.3,2F3.0,2E10.4,F6.3)
  585 FORMAT(1H0,T2,'CYC',T7,'DC',T11,'NO',T15,'DST:KM',T22,
     &     'FUEL:LT',T31,'HCG',T36,'BATEN:KH',T48,'TIRE',
     &     T56,'AERO',T64,'VEHIN'/1X,F4.0,F3.0,F4.0,F7.2,F8.3,F6.2,
     &     3F10.6,F9.6//1X,T3,'RE OUT',T12,'RE IN',
     &     T21,'IFENG:K-H',T33,'EEFF %',T42,'L/100KM',T53,
     &     'KM/L',T63,'BEOUT'/1X,3F10.6,F7.2,F10.3,F9.2,F12.7//)
  590 FORMAT(1X,2F6.1,3F8.6,F8.4,F8.4,5F8.4,F6.2,F8.4,F7.1,F8.4,F7.1)
  600 FORMAT(A1,'SUMMARY RUN RESULTS:'//
```

```
     &     1H ,T3,'TIME',T11,'NCYC',T20,'DISTANCE',T35,'TCHEM E',
     &     T46,'ELEC E',T56,'TECONS',T65,'TEDELIV'
     &     /1H ,T3,'(SEC)',T18,'(MI)',T27,'(KM)',T35,'(KW-HR)',
     &     T45,'(KW-HR)',T56,'(KW-HR)',T65,'(KW-HR)'/1H ,F8.1,I4,
     &     F9.2,F9.2,4F10.5)
630  FORMAT(1H0,T2,'SYSTEM',T10,'TCHEM E',T18,'ELEC E',T26,'TC EFF',
     &    T36,'TC EFF',T45,'TC EFF',T55,'ELC EFF',T65,'ELC EFF'/1H ,T2,
     &     'EFF (%)',T12,'(%)',T20,'(%)',T27,'(MPG)',T36,'(KM/L)',T45,
     &    'L/100KM',T55,'M/KW-HR',T64,'KM/KW-HR'/1H ,2F8.3,F7.2,2F8.2,
     &     3F10.4)
640  FORMAT(1H0,'SYS EFF  SYS EFF  SYS EFF',T31,'EBOUT',T41,'EBIN',
     &     T49,'TIRE RR  AERO  DG',T67,'ERGEN'/1H ,'M/KW-HR',
     &     T11,'KM/KW-HR  M/EQGAL  (KW-HR)  (KW-HR)  (KW-HR)',
     &     T57,'(KW-HR) (KW-HR)',/1X,F8.3,F9.4,F9.3,F10.5,1X
     &     ,3F9.4,F9.3)
650  FORMAT(1H0,' REIN',T12,'TEENG',T21,'FINAL INITL CH TIME EEFF',
     &     T48,'MEFF',T55,'BEFF',T61,'GEFF',T67,'BRAKE'/1H ,'(KW-HR)',
     &     T12,'(KW-HR)  BCHG  BCHG',T34,'(HRS)  (%)',T48,'(%)',
     &     T55,'(%)',T62,'(%)',T68,'(KH)'/1X,2F9.5,2F6.3,F7.3,3F7.2,
     &   F6.2,F6.3)
620  FORMAT(1H1,4X,'TIME',T13,'VEL',T20,'ACCEL',T27,'PTRAN',
     &     T36,'FO',T40,'WENG',T49,'PENG',T59,'EENG',T67,'TC EN',
     &      T78,'PMTR',T89,'WMTR',T97,'EMTR',T102,'MG',T105,'BC',T109,
     &       'PDS',T119,'EBAT',T128,'BCHG'/)
660  FORMAT(1H ,'ANOTHER RUN INPUT Y OR N:')
670  FORMAT(1H0,1X,'EPWR  ESPD  MPWR  GPWR  MSPD  PBOUT PBIN  ',
     &    'PDS   -PDS  VMAX   AMAX UPS'/1X,F6.1,F6.0,2F6.1,F6.0,3F6.1,
     &     F7.1,F6.1,F6.2,I3)
680  FORMAT(2X,'EPF   ESF   MPF   GPF   MSF',T37,'ITER      MON',
     &     T52,'EON  AVEPF AVESF'/1X,5F6.2,3I8,2F6.1)
      RETURN
      END
```

```
10          SUBROUTINE ACC
20C
30C       THIS SUBROUTINE COMPUTES THE ACCESSORY POWER REQUIREMENTS
40C       INCLUDE HYPR1
50C       INCLUDE HYPR2
60C
70C       ACTYP=1. STANDARD MODEL
80C       ACTYP=2. VARIABLE SPEED
90C
100C       ALL POWER IN KILOWATTS
110     50 WFA=WF/100.
120        PALT=0.0
130        PFAN=0.0
140        PPS=0.0
150        PWP=0.0
160        PAC=0.0
170        PMLOSS=0.0
180        PVLOSS=0.0
190        PSTART=0.0
200        PTLOSS=0.0
210        JAC=IAC
220        PLMT=0.0
230C
240C       CHECK AIR CONDITINER ON/OFF STATUS
250C
260 60     IF(JAC.EQ.0) GO TO 70
270        IF(ITER.GT.ITCD) GO TO 65
280        JAC=IAC
290        GO TO 70
300 65     ITEL=(ITER-ITCD)/(IAOF+IAON)
310        INEW=(ITER-ITCD)-ITEL*(IAON+IAOF)
320        IF(INEW.LE.IAOF) JAC=0
330        IF(INEW.GT.IAOF) JAC=IAC
391C/
400C       COMPUTE CLUTCH LOSS FOR FOUR SPEED TRANSMISSON
401C
410    70  IF(JELEC.EQ.1.AND.EON.EQ.0) GO TO 78
411        IF((EON+POM).EQ.0) GO TO 78
412        RDHB=.8945
420         WDS2=V*GRATS(5)*GRT*5.30516/DWHL
440        IF(MON.EQ.0) GOTO 73
450        IF(WDS2.GE.WIDLEM) GO TO 78
460        WMOT=WM
470        GO TO 74
480   73   IF(WDS2.GE.WIDLE(ETYP)) GO TO 78
490        WMOT=WE
491   74   IF(WDS2.EQ.0.0) GO TO 78
500        PLMT=PDS*(WMOT/WDS2-1.)
530C
540C       HEAT ENGINE STARTING POWER REQUIREMENTS
550C
560   78    EMMENG=0.0
561        IF(IVAC.EQ.1) EMMENG=.186
562        IF(IVAC.EQ.2) EMMENG=.56
570         GG=9.8
580        KE=9118.*EMMENG*PEMX/((WEMX)**2.)
590        EXPS=KE*GG*TSENG/JENG
600        PSTART =PEMX*EMMENG*(WSE/WEMX)**2./(1.-EXP(-EXPS))
610C
```

```
720C
730C       TRANSMISSION PUMPING LOSSES
740C
750C       IF IEAT=1       LOSSES ARE TO BE INCLUDED
760 90     IF(IEAT.EQ.0) GO TO 95
770        PTLOSS=1.62*WFA+2.69*WFA**3.0-2.2*WFA**2
780C
790C       MOTOR FRICTIONAL AND ROTATIONAL LOSSES
800C
810 95     IF(IEM.EQ.0) GO TO 100
820       PFL=CFF*(WM/WBASE)**NF+CWW*(WM/WBASE)**NW
830        PFLOSS=PFL /1000.
840       IF(IEM.EQ.1) GO TO 100
850        PFLOSS=CCC*(WM/WBASE)**NC /1000.
860C
870C      ALTERNATOR
880C
890    100 PALT=0.45*(1-IALT)+1.2*IALT
900        IF(ACTYP.EQ.2)GO TO 600
910C
920C      RADIATOR FAN
930    200 IF(IFAN.EQ.0)GO TO 300
940        IF(WFA.GT.0.22)GO TO 250
950        PFAN=0.005*PEMX
960        GO TO 300
970    250 IF(WFA.GT.0.87)GO TO 260
980        PFAN=(0.005+0.0923*(WFA-0.22))*PEMX
990        GO TO 300
1000   260 PFAN=0.065*PEMX
1010C
1020C      WATER PUMP
1030   300 IF(IWP.EQ.0)GO TO 400
1040       IF(WFA.GT.0.326)GO TO 350
1050       PWP=0.0019*PEMX
1060       GO TO 400
1070   350 PWP=(0.0019+0.035*(WFA-0.326)+0.038*(WFA-0.326)**2)*PEMX
1080C
1090C      POWER STEERING
1100   400 IF(IPS.EQ.0)GO TO 5
1110       PPS=0.746*WFA
1120C
1130C      AIR CONDITIONER
1140   500 IF(IAC.EQ.0)GO TO 999
1150       PAC=2.25+4.0*(WFA-0.2)-1.113*(WFA-0.2)**2
1160       GO TO 999
1170C
1180C      VARIABLE SPEED
1190C
1200C      RADIATOR FAN
1210   600 IF(IFAN.EQ.0)GO TO 700
1220       IF(WFA.GT.0.78)GO TO 650
1230       PFAN=0.0076*PEMX
1240       GO TO 700
1250   650 PFAN=(0.0076+0.045*(WFA-0.78))*PEMX
1260C
1270C      WATER PUMP
1280   700 IF(IWP.EQ.0)GO TO 800
1290       IF(WFA.GT.0.78)GO TO 750
1300       PWP=0.0038*PEMX
1310       GO TO 800
```

```
1320    750 PWP=(0.0038+0.025*(WFA-0.78))*PEMX
1330C
1340C       POWER STEERING
1350    800 IF(IPS.EQ.0)GO TO 900
1360        IF(WFA.GT.0.78) GO TO 850
1370        PPS=0.1865
1380        GO TO 900
1390    850 PPS=0.1865+0.34*(WFA-0.78)
1400C
1410C       AIR CONDITIONER
1420    900 IF(JAC.EQ.0)GO TO 999
1430        IF(WFA.GT.0.78)GO TO 950
1440        PAC=2.35
1450        GO TO 999
1460    950 PAC=2.35+1.525*(WFA-0.78)
1470    999 PACC(4)=PALT+PPS+PLMT
1471        IF(JAC.EQ.2) PACC(4)=PACC(4)+PAC
1480        PACC(1)=PACC(4)+PWP+PTLOSS+PFAN+PAC
1481        IF(JAC.EQ.2) PACC(1)=PACC(1)-PAC
1490    222 RETURN
1500        END
```

```
10C*************************************************
20C       EPA CYCLE SUBROUTINE
30C*************************************************
40C
50        SUBROUTINE EPACYC (T,V,A,TFIN,DCTYP)
60        PARAMETER NPTS=2137
70        DIMENSION VEL(NPTS),ACCEL(NPTS),TLIMIT(4),TZERO(4)
75        INTEGER DCTYP
80        DATA N/0/
90        DATA TLIMIT /505.,1372.,2137.,1372./
100        DATA TZERO /0.,505.,1372.,0./
110C
120C       TZERO   START OF 'V' DATA FOR PARTICULAR CYCLE
130C       TLIMIT  END OF 'V' DATA FOR PARTICULAR CYCLE
140C
150C DCTYP=5  TRANSIENT (0-505)
160C DCTYP=6  STABILIZED (506-1372)
170C DCTYP=7  HIGHWAY (1373-2137)
180C DCTYP=8  URBAN (TRANSIENT + STABILIZED)  (0-1372)
190C
200  50      IF (N.EQ.0) READ (25,10) (VEL(I),ACCEL(I),I=1,NPTS)
210 10     FORMAT(V)
220        N=1
230C
240C NOW COMPUTE V & A FROM T PARAMETER
250C
260 60    TFIN=TLIMIT(DCTYP-4)-TZERO(DCTYP-4)
270        IF (T.GT.TFIN) T=TFIN
280        I=IFIX (T+TZERO(DCTYP-4) + .49)
290        V=VEL(I)
300        A=ACCEL(I)
310 100    RETURN
320        END
```

```
010         SUBROUTINE DCMOTO
020C#################################################
030C
040C        INCLUDE HYPR1
050C        INCLUDE HYPR2
100C
110         CONV1=0.00046146
120         CONV2=7.04738
130C IF THIS IS NOT THE FIRST ITRATION SKIP THIS SECTION
140         IF(MOVER.EQ.1) GO TO 100
150C
160C  COMPUTE BASE QUANTITIES FOR MOTOR TO BE USED
170         PBASE=VBASE*IBASE
180         TBASE=CONV2*PBASE/WBASE
190         RBASE=VBASE/IBASE
200C COMPUTE BASE QUANTITIES FOR REFERENCE MOTOR
210         PBASE1 = VBASE1 * IBASE1
220         TBASE1 = CONV2*PBASE1/WBASE1
230         RBASE1 = VBASE1/IBASE1
240         HM = CONV1*DMOTI * WBASE1 **2/(2.*PBASE1)
250           JMTR=DMOTI*(PBASE*WBASE1**2)/(PBASE1*WBASE**2)
260         KE = AKV*WBASE1*FBASE1/VBASE1
270         KT = AKT*WBASE1*FBASE1/(VBASE1*CONV2)
280         CF = WWF1*(WBASE1**(NF))/(VBASE1*IBASE1*(WDAT**NF))
290         CW = WWDG1*(WBASE1**(NW))/(VBASE1*IBASE1*(WDAT**NW))
300C         CONVERT INPUT DATA TO PER UNIT QUANTITIES
310         RH1 = RST1/RBASE1
320         RH2 = RST2/RBASE1
330         BD1 = BBD1/VBASE1
340         BD2 = BBD2/RBASE1
350         BD3 = BBD3/RBASE1
360         RA = RRA/RBASE1
370         RF = RRF/RBASE1
380         RL = RRL/RBASE1
390         ACMAX = AACMAX/IBASE1
400         FCMAX = AFCMAX/IBASE1
410         FMAX = AFFO/FBASE1
420         FMIN = AFMIN/FBASE1
430         WF1 = WWF1/PBASE1
440         WDG1 = WWDG1/PBASE1
450         WLOGIC = PLOGIC/PBASE1
460         WCP=WWCP/PBASE1
470         FLUX=FLUXI/FBASE1
480         ARMLOW=CURLOW/IBASE1
490C       CONVERT BATTERY PARAMETERS
500         VNZ=VCNZ/VBASE
510         VNF=VCNF/VBASE
520         RNZ=RCNZ/RBASE
530         RNF=RCNF/RBASE
540          ALPHAT(1)=ALPHA(1)/VBASE
550          ALPHAT(2)=ALPHA(2)/VBASE
560          ALPHAT(3)=ALPHA(3)
570          ALPHAT(4)=ALPHA(4)
580          ALPHAT(5)=ALPHA(5)/VBASE
590          BETAT(1)=BETA(1)/RBASE
600          BETAT(2)=BETA(2)/RBASE
610          BETAT(3)=BETA(3)*IBASE
620          BETAT(4)=BETA(4)*IBASE
630          BETAT(5)=BETA(5)/RBASE
```

```
640         EIT=EII/VBASE
650C     SET INITIAL VALUES
660         SS=0.0
670         ISET=0
680         JFLUX=0
690        GO TO (10,20,30),IRTYP
700  10     E0 = ALPHAT(2)*NS
710         E1 = (ALPHAT(2)-EIT)*NS
720         E2 = -EETAT(2)*NS
730        GO TO 40
740  20     E0=VNF*NC1*NS
750         E1=0.0
760         E2=RNF*NC1*NS/NP
770C
780        GO TO 40
790  30     E0=VNZ*NC1*NS
800         E1=0.0
810         E2=RNZ*NC1*NS/NP
820  40     ILOSS = (WLOGIC )/E0
830         EBREF=E0*VBASE/NS
840         JUMP=1
850         VCHOP=E0
860         RPSAVE = 0.0
870         ALIMIT=IALB/IBASE1
880         AMIN1 = ACMAX/2.
890         AMIN2 = ACMAX/2.
900  50    IF(IBS.EQ.0) GO TO 60
910        ANTOT1=NP*NS
920        ANTOT2=NP2*NS2
930        ANPOLD1=NP
940         ANPOLD2=NP2
950         ANSOLD1=NS
960         ANSOLD2=NS2
970C ISKIP = 0 IF NO STARTING RESISTOR IS REQUIRED.
980C  ISKIP NOT EQUAL 0 IF JELEC = 2. I.E. STARTING RESISTOR IS REQUIRED
990  60     ISKIP = 0
1000         IF(JELEC.EQ.2)ISKIP = 1
1010         MODE = 1
1020         IF(JELEC.EQ.2) MODE = 2
1030         MOVER=1
1040C
1050C BEGIN EACH ITRATION HERE
1060C CONVERT INPUT FOR THIS ITRATION TO PER UNIT BASE
1070   100 RPM1 = WM/WBASE
1080         PMECH = PM*1000./PBASE
1090         IF(FLUX.EQ.0.0) FLUX=FLUX1/FBASE1
1100         JFLUX=0
1110         JUMP=0
1120         IF(IMGF.EQ.1) PMECH=-PMECH
1130         IF(WM.LT.WIDLEM)WM=WIDLEM
1140        IF(IBS.EQ.0) GO TO 110
1150        IF(SS.GT.0.7) GO TO 105
1160         IF(WM.GE.WBASE) GO TO 105
1170         NP=2.*ANPOLD1
1180        NS=ANSOLD1/2.
1190        NP2=2.*ANPOLD2
1200         NS2=ANSOLD2/2.
1210         ISET=ISET+1
1220         JSET=0
1230         IF(ISET.NE.1) GO TO 110
```

```
1240          E0=E0/2.
1250         E1=E1/2.
1260         E2=E2/4.
1270         GO TO 110
1280  105    JSET=JSET+1
1290          IF(JSET.EQ.1) E0=E0*2.
1300         IF(JSET.EQ.1) E1=E1*2.
1310         IF(JSET.EQ.1) E2=4.*E2
1320        NP=ANPOLE1
1330         NS=ANSOLD1
1340         NP2=ANPOLD2
1350         NS2=ANSOLD2
1360         ISLT=0
1370 110     CONTINUE
1380         RPM = RPM1
1390C TO PREVENT NUMERICAL PROBLEMS RPM NOT EQUAL 0
1400         IF(RPM1.LT.1.0E-4)RPM = 1.0E-4
1410         RAB = RA+RL
1420C TEST TO SEE IF IT IS TIME TO SWITCH RESISTORS
1430         RST=0.0
1440C        IF(ISKIP.EQ.1)RST=RH1
1450C        IF (ISKIP.EQ.2)RST=RH2
1460    115 RAA = RAB+RST
1470         GO TO (150,400,400),MODE
1480C
1490C MODE = 1 ARMATURE VOLTAGE CONTROL
1500C ########################################
1510C
1520C COMPUTE MOTOR TORQUES AND LOSSES
1530    150 WFL = CF*RPM**NF
1540         WDG=CW*RPM**NW
1550         WSL=0.01*ABS(PMECH)
1560         FLUX=FLUX1/FBASE
1570         CALL CORLOS(WCLL,FLUX,FBASE1,VPHI,VWCL,PHASE1)
1580         WCL=WCLL*(WBASE*RPM/WDAT)**NC
1590         WEL = WCL+WDG+WFL+WSL
1600         TE=(WEL+PMECH)/RPM
1610C
1620C COMPUTE ARMATURE CURRENT
1630  160     IARM = TE/(KT*FLUX)
1640C        PRESENT PROGRAM  SETS LIMIT ON MAGNITUDE OF ARMATURE CURRENT
1650         JTORQ=1
1660         IF(IARM.LT.ACMAX.AND.IARM.GT.ARMLOW) GO TO 168
1670         IF(IARM.LT.ACMAX) GO TO 265
1680         TTK=0.01
1690         IARM=ACMAX
1700  210    TE=KT*FLUX*IARM
1710         WELT=WEL-WSL
1720         PMECH=(RPM*TE-WELT)/(1.0+TTK)
1730         WEL=WELT+TTK*PMECH
1740         PM=PMECH*PBASE/1000.
1750         JTORQ=0
1760         GO TO 168
1770  265    IARM=ARMLOW
1780         TTK=-0.01
1790         GO TO 210
1800 168     VB=BD2+IARM
1810         IF(IARM.GT.ALIMIT)VB=BD2+BD1*IARM
1820         CEMF=KE*FLUX*RPM
1830         VCHOP=CEMF+VB+IARM*RAA
```

```
1840C COMPUTE APPROXIMATE BATTERY CURRENT
1850        ABAPR=(IARM+ILOSS)*VBASE1*VCHOP/(EBVEH*NS2)
1860        ABATT=ABAPR*UAHC/UAHC2
1870        EBATT=E0-SS*E1-ABATT*E2
1880        EBAT2=EBATT*EBVEH*NS2/(EBBLE*NS)
1890C
1900C COMPUTE CHOPPER LOSSES AND FIELD CURRENT
1910 170    CALL CHOP(IARM,VCHOP,EBAT2,WCHOP,VBASE1,IBASE1,2)
1920        ATARM=IARM*ARTURN
1930        CALL FIELD (FLUX,ATARM,RTURN,IFLD,FCMAX,VPHI,VAT,IBASE1,EBASE1)
1940        WFLD=(RF*IFLD*IFLD)/EFFF
1950C
1960C COMPUTE BATTERY CURRENT - SOLVE
1970C QUADRATIC EQUATION
1980 180    WLOSS=WFLD+WLOGIC+WCHOP
1990        VCHOP=KE*RPM*FLUX+VB+RAX*IARM
2000         PBAT2=WLOSS+VCHOP*IARM
2010 111     CALL BAT(1)
2020C  IF CHOPPER VOLTAGE IS GREATER THAN BATTERY VOLTAGE SWITCH TO
2030C  MODE 2 OPERATION - FIELD WEAKENING
2040C ##################################
2050        IF(VCHOP.GT.EBAT2)MODE = 2
2060        IF(ISKIP.EQ.0)GOTO 380
2070        IF(ISKIP.EQ.1.AND.IARM.LT.AMIN1)ISKIP=2
2080        IF(ISKIP.EQ.2.AND.IARM.LT.AMIN2)ISKIP=0
2090   380  IF(MODE.EQ.1)GO TO 800
2100C FIELD WEAKENING CONTROL MODE = 2
2110C ##################################
2120C COMPUTE MOTOR LOSSES FOR GIVEN SPEED
2130    400 WDG=CW*RPM**NW
2140        WFL=CF*RPM**NF
2150        WSL=0.01*ABS(PMECH)
2160C  INCLUDE BRUSH DROP
2170    420 IF(IARM.GT.ALIMIT)GOTO425
2180        VB=0.0
2190        RAX =RAA+BD3
2200        IDD = 1
2210        GO TO 430
2220    425 VB = SIGN(BD2,IARM)
2230        RAX = RAA+BD1
2240        IDD=2
2250   430  TS=(PMECH/RPM)
2260C  COMPUTE FLUX -- SOLVE QUADRATIC EQUATION
2270C  IF IN MODE 3 OPERATION SKIP THIS SECTION
2280        IF(MODE.EQ.3) GOTO450
2290 440    AA=KE*KT*RPM*RPM
2300        BB=CCOEF*RPM**NC*(RAX+E2)
2310       & +(ILOSS*E2+VB-E0+E1*SS)*RPM*KT
2320        CC=(RAX+E2)*(WFL+WSL+WDG+TS*RPM)
2330        FLUX=(-BB+SQRT(BB*BB-4.*AA*CC))/(2.*AA)
2340        JND=1
2350        TMP=BB*BB-4.*AA*CC
2360        IF(TMP.GT.0.0) GO TO 233
2370        WRITE(6,888)ITER,IARM,ABAT2,PBAT2,E0,E2,E1,EBAT2,SS,ILOSS,VB
2380       &,AA,BB,CC,CCOEF,NP,NS,NP2,NS2,PE,WE,PDS,PMXM,PMXE,V,IGEAR
2390 888     FORMAT(//,'ITER,IARM,ABAT2,PBAT2,E0,E2,E1,EBAT2,SS,ILOSS,VB',
2400       &/,'AA,BB,CC,CCOEF,NP,NS,NP2,NS2,PE,WE,PDS,PMXM,PMXE,V,IGEAR=',/,
2410       &I5,4(6E12.5/),I5)
2420 233    IF(FLUX.GT.FMIN) GO TO 442
2430        FLUX=FMIN
```

```
2440          GO TO 450
2450   442    GO TO (411,412),JELEC
2460  411     IF(FLUX.LT.FMAX. GO TO 445
2470          MODE=1
2480          FLUX=FMAX
2490          GO TO 150
2500  412     IF(FLUX.LT. FMAX) GO TO 445
2510          JFLUX=1
2520          FLUX=FMAX
2530        CEMF=KE*RPM*FLUX
2540        IF(CEMF.GT.(E0+VB)) GO TO 447
2550          JUMP=1
2560          IARM=0.0
2570          GO TO 445
2580  447     IARM=-(CEMF-E0-VB)/(E2+RAX)
2590C   COMPUTE NEW CORE LOSS COEFFICIENT
2600  445     CALL COPLOS (WCLL,FLUX,FBASE1,VPHI,VWCL,PBASE1)
2610          WCL=WCLL*(WBASE*RPM/WDAT)**NC
2620          CCOEF=WCLL*((WBASE/WDAT)**NC)/FLUX
2630C   COMPUTE ARMATURE AND BATTERY CURRENTS
2640          IF(JFLUX.EQ.0) GO TO 450
2650          WELT=WDG+WFL+WCL-WSL
2660          TE=KT*FLUX*IARM
2670          TS=TE+WELT/RPM
2680          PM=(TS*RPM)*PBASE/1000.
2690    450   IF(JFLUX.NE.1)IARM =(WDG+WFL+WSL+WCL+TS*RPM)/(RPM*KT*FLUX)
2700C   COMPUTE APPROXIMATE BATTERY CURRENT
2710   451    ABAPR=IARM+ILOSS
2720C   COMPUTE TOTAL ELECTRICAL LOSSES
2730          ATARM=ARTURN*IARM
2740          CALL FIELD (FLUX,ATARM,FTURN,IFLD,FCMAX,VPHI,VAT,IBASE1,FBASE1)
2750          WCHOP=ACP
2751          WFLD=RF*IFLD*IFLD/FEFF
2760          WLOSS=(RF*IFLD*IFLD/FEFF)+WCHOP+WLOGIC
2770          JND=2
2780          PBAT2=WLOSS+IARM*VCHOP
2790          CALL BAT(1)
2800  255     ILOSS=ABAT2-IARM
2810          VCHOP=(E0-SS*E1-IBATT*E2)*EBVEH*NS2/(NS*EBREF)
2820  475     IF(MODE.EQ.3)GOTO510
2830C   CHECK TO SEE IF BRUSH MODEL IS VALID
2840          TMP=(1.0-IARM/ALIMIT)
2850          IF(ABS(TMP).LT.TOLR)GOTO500
2860          IF(IDO.EQ.1.AND.IARM.GT.ALIMIT)GOTO425
2870          IF(IDO.EQ.2.AND.IARM.LT.ALIMIT)GOTO420
2880C   CHECK TO SEE IF ERROR IN COMPUTATIONAL IS WITHIN LIMITS
2890    500   TMP=ABS(1.0-ABAPR/ABAT2)
2900          TMX=ABS(ABAT2)
2910          TMZ=ABS(IARM)
2920          IF(TMZ.LT.0.02) GO TO 222
2930          IF(TMX.LT.0.02) GO TO 222
2940  520     CONTINUE
2950          IF(TMP.GT.TOLR)GOTO440
2960  222     CALL BAT(0)
2970          VCHOP=EBAT2
2980   510    CONTINUE
2990    800   PMI=PBASE*(VCHOP*IARM)+(IFLD**2)*RF/(FEFF*1000.)
3000          IF(IMGF.GT.0) GO TO 825
3010          OAR(1,193)=OAR(1,193)+PM*DTP/3600.
3020          OAR(3,193)=OAR(3,193)+PMI*DTP/3600.
```

```
3030          IF(INCL.EQ.0)GO TO 850
3040  825     OAR(2,193)=OAR(2,193)+PM*DTP/3600.
3050          OAR(4,193)=OAR(4,193)+PMI*DTP/3600.
3060  850     WMOTO=WCHOP*IARM+WFLD
3070          WRES=(RA+RL)*IARM**2
3080          WRST=RST*IARM**2
3090          WEBAT2=EBAT2*VBASE
3100           WABAT2=ABAT2*IBASE
3110          WPBAT2=PBAT2*PBASE
3120          WIARM=IARM*IBASE
3130          WFLUX=FLUX*FBASE
3140          WWFLD=WFLD*PBASE
3150          WWDG=WDG*PBASE
3160          WWFL=WFL*PBASE
3170          WWCHOP=WCHOP*PBASE
3180          WWRES=WRES*PBASE
3190          WWCL=WCL*PBASE
3191          WMOT=EBAT2*IARM*PBASE+WFLD*PBASE
3192          EM=PM*1000./WMOT
3193          EMT=PM*1000/WPBAT2
3194          WPM=PM*1000.
3200C   COMPUTE THE ELECTRICAL LOSSES
3210          FACTOR = PBASE*DT/3.6E6
3220          ENCL=ENCL+WCL*FACTOR
3230          ENF=ENF+WFL*FACTOR
3240          ENWDG=ENWDG+WDG*FACTOR
3250          ENSL=ENSL+WSL*FACTOR
3260          ENCHOP=ENCHOP+WCHOP*FACTOR
3270          ENFLD=ENFLD+WFLD*FACTOR
3280          ENRES=ENRES+WRES*FACTOR
3290          ENRST=ENRST+WRST*FACTOR
3300          EOUT=EOUT+PMECH*FACTOR
3310          EIN=EIN+ABAT2*EBAT2*FACTOR
3320          EIPCU=EIPCU+WCHOP+WLOGIC
3330          EIMOTO=EIMOTO+WMOTO*FACTOR
3340  999     RPSAVE=RPM
3350            RETURN
3360          END
3370C
3380          SUBROUTINE FIELD(FLUX,ATARM,FTURN,FLDI,FCMAX,VPHI,VAT,CBS,FBS)
3390  *
3400  *       PURPOSE
3410  *          TO FIND FIELD CURRENT FOR GIVEN ARMATURE CURRENT AND FLUX
3420  *
3430  DIMENSION VPHI(1),VAT(1)
3440  LOGICAL NEG
3450  DATA TOL /.005/ , NITER /20/
3460  *
3470          FILD=5.0
3480          FLX=FLUX*FBS
3490          ATAM=ATARM*CBS
3500          FFCMAX=FCMAX*CBS
3510  NEG=.FALSE.
3520  DFILD=FILD
3530  NSW=0
3540  10 DO 50 IT=1,NITER
3550  ATFLD=FILD*FTURN
3560  PHIL=.6667*PHI(ATFLD,VPHI,VAT)+.1667*(PHI((ATFLD+ATAM),VPHI,VAT)
3570  &   + PHI((ATFLD-ATAM),VPHI,VAT))
3580  IF (NSW .EQ. 0) GO TO 20
```

```
3590 FLX=PHIL
3600 GO TO 70
3610 20 ERR=FLX-PHIL
3620 IF (ABS(ERR).LT.TOL) GO TO 60
3630 IF (ERR.LT.0.) GO TO 40
3640 IF (NEG) DFILD=DFILD/2.
3650 FILD=FILD+DFILD
3660 GO TO 50
3670 40 NEG=.TRUE.
3680 DFILD=DFILD/2.
3690 FILD=FILD-DFILD
3700 50 CONTINUE
3710 *
3720 PRINT 6000,NITER,FILD
3730 6000 FORMAT(" ***ERROR***   MAX NUMBER OF ITERATIONS HAS BEEN EXCEEDED"//
3740 &    5X,"NITER=",I3,3X,"FILD=",F7.3)
3750 STOP
3760 60 IF (FILD .LT. FFCMAX) GO TO 70
3770 NSW=1
3780 FILD=FFCMAX
3790 GO TO 10
3800 70     FLDI=FILD/CBS
3810        RETURN
3820 END
3830
3840
3850 FUNCTION PHI(AT,VPHI,VAT)
3860 *
3870 *       PURPOSE
3880 *          TO FIND THE NO LOAD FLUX GIVEN THE NUMBER OF AMPERE-TURNS
3890 *
3900 DIMENSION VPHI(1),VAT(1)
3910 *
3920 PHI=0.
3930 IF (AT.LE.0.) RETURN
3940 DO 10 I=1,9
3950 J=I
3960 10 IF (AT.LE.VAT(J+1)) GO TO 20
3970 *
3980 *  AT HAS BEEN BOUNDED - COMPUTE NEW FLX
3990 20 SLOPE=(VPHI(J+1)-VPHI(J))/(VAT(J+1)-VAT(J))
4000 PHI=VPHI(J)+SLOPE*(AT-VAT(J))
4010 RETURN
4020 END
4030
4040 SUBROUTINE CHOP(IARM,VA,VBAT,WCHOP,VBASE1,ABASE1,ICHOP)
4050 REAL IARM,IA,IP,IRR,IP2,IRR2
4060        IF(ICHOP.EQ.2) GO TO 200
4070 TAU=VA/VBAT
4080 IA=ABS(IARM)*ABASE1
4090        VBT=VBAT*VBASE1
4100 IP=IA*40.
4110 IRR=.2*IP
4120 F=2000.-8000.*(TAU-.5)**2
4130 IP2=IP*IP
4140 IRR2=IRR*IRR
4150 PR7=1.0E-06*F*(IP2+IRR2)
4160 PR4=.07E-06*F*VBT*VBT
4170 P5b=5.
4180 PD9=40.E-06*F*(IP+IRR)
```

```
4190 IF (IARM .LT. 0.) GO TO 200
4200 PR1=.5E-06*F*(VBT+.26*IP)**2
4210 PR3=.5E-06*F*(VBT+.42*IRR)**2
4220 PQ1=IA*TAU+1.25E-06*F*(VBT*IP+.26*IP2)+10.*TAU
4230 PQ3=0.
4240 PD3=IA*1.35*(1.-TAU)
4250 PDA=0.
4260 GO TO 500
4270 200 PR1=.5E-06*F*(VBT+.26*IRR)**2
4280 PR3=.5E-06*F*(VBT+.42*IP)**2
4290 PQ1=0.
4300 PQ3=2.*IA*(1.-TAU)+2.5E-06*F*(VBT*IP+.42*IP2)+10.*(1.-TAU)
4310 PD3=0.
4320 PDA=IA*1.25*TAU
4330 500 PR2=PR1
4340 PR5=PR4
4350 PR6=PR4
4360 PQ2=PQ1
4370 WCHOP=PR7+PR1+PR2+PR3+PR4+PR5+PR6+PQ1+PQ2+PQ3+PD3+PDA+PSB+PD9
4380       WCHOP=WCHOP/(VBASE1*ABASE1)
4390       GO TO 1000
4400C   USE SIMPLER FORMULA IF ICHOP=2
4410 600  CCP=50./(VBASE1*ABASE1)
4420      BCP=1.4/VBASE1
4430       ACP=0.01*ABASE1/VBASE1
4440       WCHOP=ACP*(IARM)**2+BCP*IARM+CCP
4450 1000 RETURN
4460 END
4470       SUBROUTINE CORLOS(WCLL,FLUX,FBASE1,VPHI,VWCL,PBASE1)
4480       DIMENSION  VWCL(1),VPHI(1)
4490     FLX=FLUX*FBASE1
4500     IF(FLX.LT.VPHI(8))GO TO 10
4510       WCLL=VWCL(8)/PBASE1
4520       GO TO 999
4530 10   DO 851 I=1,9
4540 J=I
4550 851 IF(FLX.LE.VPHI(J+1)) GOTO 852
4560 852 DWCL=VWCL(J+1)-VWCL(J)
4570 DPHI=VPHI(J+1)-VPHI(J)
4580 WCL1=VWCL(J)+(FLX-VPHI(J))*DWCL/DPHI
4590 990   WCLL=WCL1/PBASE1
4600  999 RETURN
4610       END
```